Universidade Vigo

EIDO Escola Internacional de Doutoramento

Carlos Groba Presa

TESIS DOCTORAL

Essays in logistics optimization: Algorithms and Game Theory for solving the Traveling Salesman Problem in dynamic scenarios

Dirigida por los doctores:
Prof. Xosé Henrique Vázquez Vicente
Prof. Antonio Sartal Rodríguez

2018

Universidade Vigo

EIDO
Escola Internacional
de Doutoramento

UniversidadeVigo
Escuela Internacional de Doctorado

Carlos Groba Presa

TESIS DOCTORAL

Essays in logistics optimization: Algorithms and Game Theory for solving the Traveling Salesman Problem in dynamic scenarios

Dirigida por los doctores:
Prof. Xosé Henrique Vázquez Vicente
Prof. Antonio Sartal Rodríguez

2018

Agradecimientos

Si echo la vista atrás, durante el largo camino recorrido, son muchas las personas a las que tengo que agradecer su tiempo, amabilidad y aliento a lo largo de todos estos años. Como la lista es larga, quiero pedir disculpas de antemano por si me olvido de alguien que debería estar aquí.

En primer lugar me gustaría agradecer a mis padres todo su esfuerzo. Sin su sacrificio y apoyo incondicional todo esto sería imposible. También a mi hermano Berto por su apoyo durante todos estos años e incluso ayudarme en algunos aspectos de este trabajo.

Debo agradecer el origen de este proyecto a Xosé H. Vázquez. Al finalizar el máster de Innovación y Optimización de Procesos, que él mismo dirigía, me planteó la posibilidad de hacer la tesis, cosa que a mí nunca se me había pasado por la cabeza. A partir de ese momento todo cambió y nuestros caminos se unieron. A Kike le tengo que agradecer en primer lugar su confianza en mí, su motivación, liderazgo y ayuda como director, pero también su cercanía y sus consejos a nivel personal. Todo esto ha hecho que nuestra relación se convirtiese en una amistad en la que nos propusimos hacer una tesis y divertirnos por el camino. Kike, gracias por todo.

En el máster de Innovación, tuve la suerte de tener grandes compañeros, entre los que quiero destacar a Manu, Eva, Mercedes y Diego, entre muchos otros. Iago y Bastián, en particular, compartirían conmigo la senda de la investigación más adelante. Durante este camino también quiero mencionar a Javier Hervés, compañero en el Máster de Economía con el que, junto a Iago y Bastián, tantas horas compartimos trabajando, pero sobre todo disfrutando de lo que hacíamos en un ambiente inmejorable.

Debo nombrar también a Marcos Álvarez, profesor de la UVigo, que me ayudó en los primeros pasos cuando más orientación necesitaba.

Quiero mostrar mi agradecimiento también y especialmente a mi co-director de la tesis, Antonio Sartal. Su ayuda ha sido crucial para que este trabajo pudiese continuar cuando estaba en sus horas más bajas. Toño vino para revolucionar y dinamizar la forma de trabajar. A partir de su llegada todo ha sido más fácil y hemos disfrutado juntos de muchísimas jornadas de trabajo. Quiero agradecerle su entusiasmo, optimismo, y su capacidad para convertir el trabajo en placer. Al igual que Kike, Toño es un amigo, y es una gran suerte tenerlos a mi lado.

No quiero olvidar a los compañeros del departamento que también están con la aventura de la tesis entre manos y con los que he compartido muy buenos momentos, son: Iago, Luis y Diego. A todos ellos les quiero dar las gracias e intentar motivar con la presentación de este trabajo.

Quiero agradecer también a los compañeros de la Facultad de Ciencias Económicas y Empresariales, Ana, María, Manolo y Javi por ser tan amables conmigo y por su constante interés por mis avances. También a Carlos Hervés por su disponibilidad para cualquier duda y a Álvaro Dosil, por su ayuda en la maquetación del trabajo.

Mención especial merece la empresa *Marine Instruments*, en la que trabajo desde 2005. Sin duda ha sido el caldo de cultivo perfecto para fomentar mi curiosidad día tras día. Esta tesis no hubiese transcurrido de la manera en que lo ha hecho si no hubiese estado trabajando en *Marine*, pues eso me llevó a pensar en una solución para la ruta óptima de los atuneros, primer paso de la investigación. Además, sin los datos de las boyas, tan amablemente cedidos por la empresa, este trabajo no habría sido posible.

Dentro de *Marine Instruments* quiero agradecer especialmente a su fundador, Francisco Pino, todo el aprendizaje que he recibido de él durante todos estos años. Su creatividad sin parangón, curiosidad, generosidad, inteligencia pero a la vez modestia, siempre han sido y son un referente para mí, como profesional y como persona.

Quiero mencionar también a todos los compañeros de la empresa que me han ayudado durante estos años, ya sea a través de su paciente escucha como a través de diferentes contribuciones a lo largo del tiempo, entre los que quiero destacar a Iago Paz y Bruno Lema.

Finalmente quiero hacer mención especial a Marina. Por su ánimo y apoyo constante y por soportar tantas horas mi ausencia, tanto física como de pensamiento, durante la evolución de este trabajo.

A todos ellos mi más sincero agradecimiento.

Carlos

A Marina

Índice general

Índice de figuras **vii**

Índice de cuadros **ix**

1 Introducción **1**
1.1 Objetivos . . . 1
1.2 Discusión general . . . 2
1.3 Conclusiones . . . 7

2 Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices **9**
2.1 Abstract . . . 11
2.2 Introduction . . . 11
2.3 Literature review . . . 12
2.4 The tuna vessel FAD recovery problem . . . 13
2.5 Drifting object prediction . . . 14
2.5.1 Time series . . . 15
2.5.2 Newton's motion equation . . . 16
2.5.3 Model comparison . . . 17
2.6 Methodology . . . 19
2.6.1 Data on buoys and vessels . . . 19
2.6.1.1 Buoys input . . . 19
2.6.1.2 Vessel and fishing information input . . . 20
2.6.1.3 Objects prediction . . . 20
2.6.2 Genetic Algorithm design . . . 21
2.6.2.1 Solution encoding . . . 21
2.6.2.2 Initialization . . . 21
2.6.2.3 Selection . . . 21
2.6.2.4 Crossover . . . 22
2.6.2.5 Mutation . . . 22
2.6.2.6 Stopping Criteria . . . 22
2.6.3 GATP final solution: implementing the GA with Trajectory Prediction . . . 23
2.7 Results and discussion . . . 27
2.8 Conclusions . . . 29

3 Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels **31**
3.1 Abstract . . . 33
3.2 Introduction . . . 33
3.3 Literature Review . . . 34

3.4 Data . . . 36
3.4.1 Introduction: FAD recovery for tuna vessels . . . 36
3.4.2 Data on buoys and vessels . . . 38
3.4.2.1 FADs input . . . 38
3.4.2.2 Vessels and fishing-information input . . . 39
3.5 Methodology . . . 39
3.5.1 The predictive technique: estimating FADs future position . . . 39
3.5.2 Algorithm design . . . 41
3.5.2.1 Solution encoding . . . 43
3.5.2.2 Initialization . . . 43
3.5.2.3 Prediction . . . 44
3.5.2.4 Fitness . . . 44
3.5.2.5 Selection . . . 44
3.5.2.6 Crossover . . . 44
3.5.2.7 Mutation . . . 45
3.5.2.8 Execution of the algorithm and stopping criterion . . . 45
3.6 Implementing the GA with a multiple-trajectory prediction . . . 46
3.7 Computational results . . . 48
3.8 Conclusions . . . 52
3.A Appendix . . . 54

4 Optimization of logistic routes through information sharing policies: A game theory-based approach **55**
4.1 Abstract . . . 57
4.2 Introduction . . . 57
4.3 Background . . . 58
4.4 The tuna fishing vessels problem: A game theory approach . . . 60
4.4.1 The tuna fishing vessels problem . . . 60
4.4.2 The tuna fishing vessels problem . . . 60
4.4.3 A game theory approach . . . 61
4.5 Data and results . . . 64
4.6 Conclusions . . . 67
4.A Appendix . . . 69
4.A.1 Mechanism 1. Reassigning FADs without compensation . . . 69
4.A.2 Mechanism 2. Reassigning FADs with compensation . . . 75

5 Conclusiones **79**
Bibliografía . . . 81

Índice de figuras

2.1 FADs drifting in the Indian Ocean 15
2.2 Two FADs drifting in the ocean 16
2.3 GATP Block Diagram 23
2.4 GATP Fitness calculation 24
2.5 Using GA based combining prediction methods 27
2.6 NN, GA-TSP and GATP: A graphic example for 12 FADs 27

3.1 Drifting FADs and three vessels in the Indian Ocean. 38
3.2 GAMTP algorithm procedure. 49
3.3 GAMTP improvements evolution as the ratio of FADs/vessel increases (built from Table 2). 51

Índice de cuadros

2.1 Buoy data samples 18
2.2 MAPE for three prediction methods 19
2.3 Results comparison 28
2.4 Repeated Measures ANOVA: multivariate tests 28
2.5 Repeated Measures ANOVA: Pairwise Comparison 29

3.1 Mathematical symbols used. 38
3.2 Computational results. 50
3.3 Multivariate test (ANOVA) 50
3.4 Pairwise comparison (ANOVA) 51
A1 Experiment design 54

4.1 Experiment assumptions 64
4.2 Current way of working: Vessels do not share their FADs 65
4.3 Mechanism 1: Reassigning FADs without compensation 66
4.4 Mechanism 2: Reassigning FADs with compensation 67

1 Introducción

Esta Tesis Doctoral presenta tres ensayos que abordan problemas de optimización logística con una característica especial: se producen en entornos dinámicos. Esta clase de problemas representan un ámbito de estudio crecientemente importante porque son cada vez más frecuentes en las operaciones empresariales y añaden cierta complejidad analítica a los típicos problemas de entornos estáticos. Por este motivo, y manteniendo el énfasis en la generalidad del problema, no en la resolución de un reto empresarial concreto, en este trabajo se desarrollan técnicas y algoritmos que avanzan el conocimiento disponible con un enfoque multidisciplinar.

El camino se inicia en el escenario más básico dentro un entorno dinámico, que es el que estudia el primer trabajo. En el segundo, se amplia la dificultad del problema con un escenario más generalista y a la vez más complejo. Finalmente, en el tercer trabajo, se realiza un estudio todavía más amplio, que concluye con recomendaciones sobre las políticas que las empresas deben seguir para ser más eficientes y, por lo tanto, más competitivas.

El primero de los ejercicios ha dado lugar a una publicación en el journal *'Computers & Operations Research'*, en el año 2015 y coautoría de Antonio Sartal y Xosé H. Vázquez.

El segundo de los ejercicios, también en coautoría de Antonio Sartal y Xosé H. Vázquez, ha motivado una publicación en el journal *'Engineering Applications of Artificial Intelligence'*, en el año 2018.

El tercer ejercicio, en coautoría de Antonio Sartal y Gustavo Cid Bergantiños, ha sido enviado al journal *'Marine Policy'* en Septiembre de 2018, encontrándonos en estos momentos a la espera de respuesta.

1.1 Objetivos

El objetivo general de los trabajos que engloban esta Tesis Doctoral es el de mejorar la optimización logística en entornos dinámicos. De manera particular, en cada uno de los trabajos se desarrollarán algoritmos y técnicas que permitan incrementar la eficiencia en cada caso donde se apliquen. Los resultados esperados son, por lo tanto, una mejora de la eficiencia demostrable en cada uno de los escenarios propuestos, que serán contrastados en todos los casos con datos reales y comparados con otras técnicas.

1.2 Discusión general

El mundo de la logística ha sido ampliamente abordado desde hace décadas, pero el movimiento de flotas y mercancías ocupa un espacio cada vez mayor que va paralelo a la globalización de la economía y al cambio tecnológico. El hecho es que la logística mundial no ha parado de crecer en los últimos años, y el número y volumen de mercancías que se mueven está en continuo aumento, ya sea por tierra, mar o aire. Se generan así constantemente nuevos y numerosos retos para las empresas que, de manera sintética, podrían resumirse en dos grandes preocupaciones: de un lado, cumplir con los plazos de respuesta acordados gestionando cada vez un mayor número de referencias y restricciones de entrega; de otra parte, hacer esto con la menor utilización de recursos posible. Los problemas de optimización logística como los que abordamos aquí tienen que ver de alguna u otra forma con ambos aspectos.

Esta Tesis Doctoral aborda el tipo de problemas que surgen a la hora de hacer que el movimiento de flotas y mercancías sea óptimo. Aunque la amplitud del problema es considerable, uno de los retos a resolver más relevantes es el de obtener la mejor ruta a realizar. Para resolver este problema es necesario encontrar la solución de un problema de optimización. Sin embargo, encontrar esta solución a menudo es una tarea compleja. La razón es que, en muchas ocasiones, los problemas logísticos son del tipo NPC (*Non-deterministic Polynomial time Complete*), lo cual significa que el tiempo requerido para resolver el problema usando cualquier algoritmo aumenta muy rápido según crezca el tamaño del problema. Esta característica convierte frecuentemente este tipo de retos en problemas inabarcables computacionalmente incluso para los ordenadores de hoy en día.

El problema del viajero o TSP (*Traveling Salesman Problem*) es uno de los problemas NPC más conocidos, donde se trata de calcular la ruta óptima que un viajero debe realizar para recorrer n ciudades. La complejidad de un problema NPC es fácil de visualizar en el caso del TSP. Por ejemplo, si el número de ciudades a recorrer es solamente 10, el viajero tendrá que calcular $3,6 \times 10^6$ de combinaciones para encontrar la mejor solución, pero si el número de ciudades aumenta a 100, el número de combinaciones se dispara a $9,3 \times 10^{157}$. Este sencillo ejemplo muestra el rápido aumento en combinaciones a calcular (lo hace con n) cuando el tamaño del problema crece de manera lineal, de ahí que sea extremadamente difícil resolver este tipo de problemas.

La importancia del TSP, además de su complejidad, radica en su versatilidad para ser adaptado a otros escenarios más allá de la logística. El concepto de ciudades se puede transformar en clientes, puntos de soldadura, fragmentos del ADN, etc. Mientras que el concepto de distancia puede transformarse en cualquier variable que implique un coste. De esta forma su resolución puede usarse para adaptar problemas de diferentes ámbitos como las finanzas, medicina, ingeniería, etc.

El TSP fue enunciado por primera vez en 1930 y desde entonces es uno de los problemas de optimización más estudiados. Existen multitud de procedimientos para intentar encontrar las mejores soluciones, ya sea de manera exacta (si el tamaño del problema lo permite) o a través de aproximaciones o heurísticas, que tratan de reducir el espacio de búsqueda asegurando llegar a soluciones relativamente buenas en menor tiempo. A pesar de la importante evolución que ha tenido el estado del arte para resolver el TSP, las soluciones alcanzadas pierden efectividad y son difícilmente aplicables cuando el entorno donde se producen es dinámico.

En efecto, las mejores rutas que se puedan calcular no lo son tanto ante un entorno de cambio constante. Esto es debido a que la solución se calcula para un escenario estático, y cualquier variación sobre sus condiciones conlleva, en la mayoría de los casos, a que la solución deje de ser eficaz. Aunque un escenario dinámico pueda parecer un entorno de corte teórico, lo cierto es que no lo es en absoluto. En efecto, la realidad del mundo en el que vivimos es que se producen cambios de manera constante y gran parte de los problemas a resolver se producen en escenarios dinámicos. Como ejemplo, y para el caso concreto de la logística, pedimos algo por internet, pero al día siguiente modificamos el pedido añadiendo o sacando nuevos productos, cambiando la cantidad, etc. Esto sucede desde la perspectiva de un único cliente. Si lo extendemos a miles y le añadimos otro tipo de variables dinámicas, como las climatológicas y del entorno, entonces el problema a resolver adquiere una complejidad inmanejable.

Entonces, ¿cómo tomar la mejor decisión en términos de optimización cuando la complejidad a la que nos enfrentamos es tan grande y a la vez difícil de calcular?

Este trabajo pone el foco en este tipo de problemas, comenzando en el análisis del problema más particular dentro del TSP en entornos dinámicos para llegar, paso a paso, a una solución más general y adaptable a otros ámbitos, que se irá mostrando según avanzamos en los ensayos.

La tesis utiliza como evidencia empírica para testar la soluciones propuestas el caso particular de los barcos atuneros que pescan con objetos a la deriva (FADs) en zonas tropicales. La razón es que esta industria se enfrenta a los mismos problemas logísticos que el TSP, pero con mayor complejidad, pues esos objetos se mueven con el tiempo en el agua, presentando un escenario totalmente dinámico. Otra de las razones para centrarnos en este caso concreto es que se disponen de datos reales provenientes de la flota, muy útiles para contrastar las soluciones planteadas en cada uno de los ejercicios y conocer el grado de mejora alcanzado para cada uno de los casos.

A continuación se detalla cada uno de los ejercicios que componen esta Tesis Doctoral:

Ensayo 1: *Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices.*

En este primer ensayo introducimos el problema de optimización logística en entornos dinámicos, conocido como DTSP (*Dynamic Traveling Salesman Problem*), mostrando la necesidad específica que un barco atunero tiene a la hora de trabajar con objetos a la deriva. En este caso el barco atunero sería el equivalente al viajero, y los objetos a recoger, el equivalente a las ciudades. La problemática que añade el DTSP respecto al TSP, es que los objetivos son dinámicos, es decir, para el caso del TSP es como si las ciudades se moviesen en el tiempo, añadiendo más complejidad al problema. En efecto, los objetos (FADs) que el atunero debe recoger para pescar se mueven. Derivan en el océano siguiendo las diferentes corrientes, complicando la toma de decisiones cuando el barco atunero plantea cuál es la mejor ruta a realizar para recogerlos y pescar en ellos.

A través de una novedosa solución, que combina sistemas predictivos con el uso de algoritmos evolutivos (algoritmos genéticos), se construye un algoritmo que asimila la componente dinámica de este escenario particular y hace que el algoritmo evolucione hasta alcanzar soluciones que permiten una mejora sustancial en comparación con otros métodos, tanto los que son usados hoy en día por los atuneros, como los que existen en el estado del arte. El algoritmo usa como parámetros las posiciones actuales y pasadas de los FADs, así como la velocidad a la que el barco se va a desplazar y el tiempo que va a estar pescando en cada objeto. A través de estas variables, combinando la predicción y el uso del algoritmo genético, se llega a la solución descrita anteriormente.

El rendimiento del algoritmo es comparado con la forma en la que los atuneros realizan la ruta de recogida hoy en día, siguiendo la técnica del vecino más cercano o NN (*Nearest Neighbour*), lo que se traduce en ir a buscar el objeto más cercano, el cual es un óptimo local, pero está lejos de ser la mejor solución global. Por otra parte también se realiza la comparación con un algoritmo evolutivo del estado del arte. De esta forma la solución planteada se compara con un algoritmo genético que no incorpore predicción, que es la novedad de este trabajo. Se observa que la solución desarrollada es mejor en ambos casos, respecto a si se compara con la técnica de recogida NN, pero también en comparación con el algoritmo genético, representando al estado del arte.

Los resultados obtenidos están contrastados con datos reales, observando que el algoritmo diseñado muestra importantes mejoras en términos de eficiencia y, por lo tanto, en el ahorro de costes al lograr mejores rutas. De manera indirecta, pero no menos importante, la solución propuesta permite que se reduzcan las emisiones de CO_2 a la atmósfera de manera significativa.

La solución, además de arrojar luz sobre los problemas de optimización en entornos dinámicos, pasa a ser una generalización de los problemas estáticos, válida para solucionar el DTSP y también el TSP.

La generalidad de este trabajo permite que el algoritmo pueda ser adaptado fácilmente a otros entornos logísticos en los que se pueda sacar partido a la solución propuesta, pero también es aplicable en otros campos diferentes a la logística, como pueden ser la medicina o la física, entre otros.

Ensayo 2: *Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels.*

En este segundo ensayo se amplía el escenario planteado en el primer trabajo, que exponía el reto de optimización de la ruta que un atunero presenta ante la recuperación de un conjunto de objetos a la deriva. En este nuevo trabajo el entorno es más completo y a su vez más difícil de resolver, pues se añade una nueva dimensión de complejidad y abstracción al problema. En este caso el reto será el de resolver el mismo problema que en el primer trabajo, pero globalmente, para una flota de barcos. La diferencia es que el número de barcos puede ser variable, desde uno (el caso que es estudiaba con anterioridad) hasta un número

m, tratando de conseguir una optimización de la ruta recorrida por la flota de manera global, en un entorno no estático, donde los objetos van a la deriva en el océano. Este ejercicio amplia el escenario planteado en el anterior trabajo a uno más realista. Esto es debido a que, en muchas ocasiones, los barcos atuneros trabajan de manera agrupada, por lo que el problema a resolver deja de ser un problema local de un solo barco, y pasa a ser un problema grupal de m barcos con n boyas o FADs (siendo $n \gg m$).

Este segundo trabajo explora el estado del arte en cuanto a este tipo de problemática, denominada MDTSP (*Multiple Dynamic Traveling Salesman Problem*), que hace referencia al problema tratado anteriormente, pero con la particularidad que hay más de un viajero. Este problema se estudia también desde el conocido VRP (*Vehicle Routing Problem*), una generalización del MTSP con aplicaciones particulares en transporte y logística, donde se pueden añadir restricciones al problema, como la distancia máxima recorrida por los agentes, la capacidad de cada uno de ellos, etc. El trabajo realizado usa la literatura del MTSP como base, pero la solución obtenida es también válida para el entorno del VRP.

De nuevo se realiza una aproximación a través de un algoritmo evolutivo que usa las estimaciones futuras de las localizaciones de los objetos a la deriva y los combina con datos de la velocidad de cada uno de los barcos y el tiempo de pesca. La combinación de todos estas variables a través del algoritmo genético permite mejorar sustancialmente sus resultados si son comparados con otros métodos del estado del arte.

Los resultados alcanzados, contrastados a través de datos reales, son prometedores para la industria atunera por el grado de ahorro en combustible y tiempo, pero igualmente aplicables en cualquier tipo de industria que se enfrente a un problema de naturaleza similar. Particularmente esta solución podría encontrar aplicación en diversos escenarios, como el movimiento de drones para transporte, cálculo de rutas en ámbito militar, u otros, pero podría ser extensible a cualquier otro tipo de aplicaciones del mundo real, desde el ámbito científico al de negocios, así como en medicina, física, producción o logística.

Ensayo 3: *Optimization of logistic routes through information sharing policies: A game theory-based approach.*

El tercer ensayo pretende cerrar el estudio de optimización a través de un desarrollo teórico que después es contrastado empíricamente, con datos reales. El trabajo versa sobre el paradigma al que la industria atunera se enfrenta en estos días, y profundiza en la evolución de la pesca industrial del atún que, con las actuales limitaciones legislativas en el uso de objetos a la deriva (FADs), debe seguir manteniéndose competitiva.

Con cada vez más restricciones por parte de los organismos reguladores, las empresas atuneras deben trabajar cada día con menos objetos en el agua. Mientras tanto deben tratar de mantener los márgenes operativos, para así poder amortizar las grandes inversiones que han realizado durante los últimos años, ya sea en la compra o construcción de barcos, licencias de pesca, etc.

Usando como base los trabajos anteriores, que ya permiten mejorar la eficiencia de los barcos que usan objetos, este último trabajo de la Tesis Doctoral se focaliza de una manera diferente. Este ejercicio realiza una comparativa entre los diferentes modos de trabajar de los barcos atuneros y realiza un novedoso estudio del comportamiento de los barcos atuneros cuando trabajan con objetos. Este estudio se realiza desde la perspectiva teórica de la teoría de juegos para demostrar por qué el entorno actual de trabajo no estimula a los barcos la compartición de objetos entre si, lo que implica una solución no óptima a nivel de optimización logística.

A través de la modelización de los intereses de los patrones, que quieren maximizar lo que pescan debido a que tienen incentivos para ello, se modelizan también los intereses de la compañía, que también trata de maximizar la pesca de cada uno de sus barcos, pero a su vez debe minimizar los costes variables, como el combustible, tripulación, pertrechos y demás.

A través de esta modelización se realiza un estudio con dos propuestas donde se observa de manera teórica que, si bien actualmente no existen incentivos para que los patrones compartan la información de sus objetos, a través de compensaciones sí podría existir un escenario donde todos ganan. Los resultados teóricos alcanzados son contrastados de manera empírica, con datos reales a través de simulaciones, sugiriendo que efectivamente existe un escenario donde todos los agentes podrían ganar: los patrones, la empresa y también el medio ambiente. Esto es debido a que, al conseguir la ruta óptima de toda la flota, también se optimiza el uso de combustible y, por lo tanto, las emisiones de CO_2 a la atmósfera. Para ello se diseñan unas políticas con compensaciones, donde los patrones obtienen los incentivos necesarios para compartir los objetos y así permitir a la empresa realizar una optimización logística global.

Los resultados muestran mejoras importantes para la flota, simplemente con el hecho de que los barcos compartan la información, pues optimizan la ruta global cubriendo áreas más específicas por barco. Así consiguen ser más eficientes.

El nuevo reparto de objetos entre toda la flota se realiza a través de la técnica del vecino más cercano (Nearest Neighbour; NN). Es salientable que esta mejora obtenida podría ser mejorada en un porcentaje importante si en el reparto se aplica el algoritmo MDTSP desarrollado en el segundo ejercicio. En ese caso la mejora alcanzada por la empresa sería todavía mayor, con más beneficios para la empresa, posibilidad de dar más incentivos a los patrones y, sobre todo, minimizar las emisiones a la atmósfera.

Este ejercicio concluye con recomendaciones claras sobre las políticas que las empresas de atuneras deberían seguir para mejorar sus resultados, fomentando la compartición de objetos entre sus barcos. El estudio teórico realizado, a la vez contrastado con datos reales, abre una puerta a una mejora de la eficiencia global, crucial en los tiempos actuales para las empresas que desean seguir siendo competitivas en un entorno tan regulado como lo es el de la pesca con objetos.

1.3 Conclusiones

En este apartado se resumen las principales contribuciones de cada uno de los trabajos.

Ensayo 1: *Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices.*

Este trabajo aborda la resolución del problema del viajero en entornos dinámicos (DTSP). Para ello se desarrolla un algoritmo genético que usa las predicciones de los objetos en movimiento. La combinación del algoritmo genético con las predicciones permite que las soluciones que progresivamente se van produciendo en cada una de las generaciones evolucionen hacia rutas quasi-óptimas, adaptadas al entorno dinámico. El algoritmo ha sido probado con datos reales para el caso concreto de los barcos atuneros en la recolección de objetos a la deriva (FADs). Los resultados muestran importantes mejoras en comparativa con otros métodos usados, ya sea por la industria o el estado del arte. La solución desarrollada, además de permitir una mejora en optimización para este caso particular, es fácilmente trasladable a otros entornos que muestren características similares. Además presenta una generalización válida tanto para entornos dinámicos como estáticos.

Ensayo 2: *Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels.*

Este trabajo se focaliza en la resolución del problema del viajero múltiple en entornos dinámicos (MDTSP). Se trata de una ampliación en complejidad del ejercicio anterior, siendo la diferencia fundamental que este problema es más generalista, con mayor aplicación en el mundo real, pues es válido para cualquier cantidad de viajeros o recolectores, mientras que en el primer ejercicio solamente lo era para uno. Para resolver este reto se desarrolla un algoritmo evolutivo que solucione el MTSP, y para la componente dinámica se usan predicciones, al igual que en el ejercicio anterior. El algoritmo desarrollado converge a soluciones del MTSP donde la optimización alcanzada es global. A través de datos reales se demuestra, para el caso de los barcos atuneros con FADs, que la solución alcanzada tiene mejores resultados que otras aproximaciones, ya sean del estado del arte o usadas actualmente por la industria. Estos resultados muestran que es posible mejorar la eficiencia en términos logísticos, con implicaciones tan importantes como el ahorro de combustible, tiempo de pesca y emisiones de CO_2 a la atmósfera. La solución alcanzada es aplicable a múltiples escenarios logísticos, pero también es válida para otros entornos. Al igual que en el ejercicio anterior, la solución es válida para entornos estáticos y dinámicos, convirtiéndola en una generalización para ambas.

Ensayo 3: *Optimization of logistic routes through information sharing policies: A game theory-based approach*

Este trabajo realiza un estudio del comportamiento de los agentes de la industria atunera tropical cuando trabajan con objetos a la deriva (FADs). Para ello se modeliza el comportamiento de los patrones de pesca y el de la compañía. El objetivo es entender desde una perspectiva matemática, en base a la teoría de juegos, el por qué de las decisiones de cada uno de los agentes toma en cada caso. El trabajo postula dos escenarios, el usado en la actualidad por la industria y uno nuevo, con compensaciones para compartir los FADs. Los resultados teóricos alcanzados muestran que existen equilibrios para la empresa y los patrones, y estos resultados son confirmados a través de datos reales cuando comparten los objetos entre toda la flota. Los resultados muestran que en el nuevo escenario propuesto, mediante incentivos y compartición de FADs, todos los agentes ganan y se produce una optimización global de los recursos. Da esta forma la ruta seguida por cada atunero se minimiza, así como el tiempo de pesca y, sobre todo, las emisiones de CO_2 a la atmósfera.

El ejercicio muestra que, mediante las políticas adecuadas, es posible mejorar la eficiencia por medio de la optimización del uso de los FADs de la compañía, haciéndola más competitiva.

2

Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices

En este capítulo nos centramos en la resolución del problema del viajero en entornos dinámicos (DTSP). Para ello se diseña una solución que combina métodos heurísticos con técnicas predictivas para obtener una solución que mejora los resultados alcanzados en entornos dinámicos, minimizando la distancia recorrida.

La solución al problema DTSP se obtiene mediante la construcción de un algoritmo genético para resolver el problema del TSP estático, pero que es modificado para el caso dinámico. Para ello se usa la ecuación de Newton como técnica predictiva, incorporando posteriormente esta información al algoritmo genético, que la usará para evolucionar sobre escenarios más realistas, permitiendo alcanzar mejores resultados.

En primer lugar se realiza una predicción de la trayectoria futura de cada uno de los objetos del problema. Después se usará esta información para que el algoritmo genético sepa evaluar correctamente cada una de las rutas alcanzadas en función de la posición estimada de cada objeto particular en función del tiempo, obteniendo resultados más realistas, que se adaptan mejor a un entorno dinámico.

Este ejercicio está basado en la evidencia empírica de los barcos atuneros que trabajan con objetos (FADs). Estos barcos pescan con objetos a la deriva, que a su vez llevan boyas de posicionamiento que envían la información vía satélite, con datos de posición (GPS), nivel de batería, información acústica, etc. Esta información es recibida a bordo del barco y por lo tanto éste se encuentra ante un problema de optimización logística, del estilo del TSP, a la hora de tomar una decisión sobre cuál es la mejor ruta para pescar. La problemática de que los objetos vayan a la deriva convierten a este problema en un reto a resolver, donde el estado del arte apenas tiene soluciones válidas.

Este trabajo tiene como resultado el diseño de un algoritmo, llamado GATP, que integra el método de predicción y la potencia de un algoritmo genético para obtener soluciones que mejoran los resultados en entornos dinámicos. Los resultados obtenidos, contrastados con datos reales provenientes de la flota atunera, muestran que la solución diseñada mejora en comparación con otros métodos usados, ya sea por la industria atunera o el estado del arte, permitiendo una mayor eficiencia de la flota atunera.

La solución propuesta, gracias a la mejora la ruta recorrida por el barco atunero, logra reducir las emisiones de CO_2 a la atmósfera de manera significativa.

Además de todo esto, la solución alcanzada permite una generalización para la problemática del TSP en entornos dinámicos y estáticos, convirtiéndose en una generalización del problema global y siendo de interés para los problemas de optimización logística de cualquier naturaleza, pero también aplicable en otros ámbitos, como puede ser la medicina, física, etc.

Este ejercicio ha dado lugar a una publicación en el journal *'Computers & Operations Research'* en el año 2015 y coautoría de Antonio Sartal y Xosé H. Vázquez.

2.1 Abstract

The paper addresses the synergies from combining a heuristic method with a predictive technique to solve the Dynamic Traveling Salesman Problem (DTSP). Particularly, we build a genetic algorithm that feeds on Newton's motion equation to show how route optimization can be improved when targets are constantly moving. Our empirical evidence stems from the recovery of fish aggregating devices (FADs) by tuna vessels. Based on historical real data provided by GPS buoys attached to the FADs, we first estimate their trajectories to feed a genetic algorithm that searches for the best route considering their future locations. Our solution, which we name Genetic Algorithm based on Trajectory Prediction (GATP), shows that the distance traveled is significantly shorter than implementing other commonly used methods.

2.2 Introduction

The Traveling Salesman Problem (TSP) probably represents the most intensive area of research within the wide range of combinatorial optimization problems (Golden et al, 1987; Gutin and Punnen, 2002). Whereas the diverse perspectives and problem-solving methods have helped practitioners and scholars to address a multitude of different problems in different industries (Grötschel and Padberg, 1985; Lawler et al, 1985; Duchenne et al, 2007; Donald, 2010), the literature on TSP is still underdeveloped with regard to moving targets -such as in the fishing or military industries (Helvig et al, 2003). In this case, the most recent approaches (which can be grouped under the heading "Dynamic Traveling Salesman Problem"-DTSP) work on a real time basis to find the changes between nodes (Pantrigo and Duarte, 2013); nevertheless, they do not anticipate the future movement of targets, so the optimal solution is given only when changes happen and the algorithm is subsequently recalculated.

With this academic background in mind, we faced the problem of tuna vessels that pick up fish aggregating devices (FADs) at sea. When FADs transmit information on how much tuna might be available beneath them, the vessels need to design a route taking into consideration that FADs are constantly moving. They need to minimize distance while recovering the FADs because saving time and fuel determines their competitiveness. Using therefore real data, the paper contributes to the literature by proposing a new approach that combines a heuristic method with a predictive technique. Particularly, we first estimate the trajectories of the FADs to subsequently build a genetic algorithm that uses this information and searches for the best possible route considering their future locations.

From all heuristic methods, we chose GAs for their properties (they are evolutionary, show statistical convergence, and tend to a global optimum with considerable robustness) and because they offer vessels the possibility to reach a solution within an acceptable computational time (Jih and Hsu, 2004; Bjarnadóttir, 2004). On the other hand, we chose Newton's movement equation as a predictive technique (we show a performance comparison with other techniques for illustrative purposes) because it offers vessels a sound and quick forecast of the future position of FADs with very little information. By combining both tools in a single method, which could be named Genetic Algorithm based on Trajectory Prediction (GATP), we reach a global optimization solution with statistically better results than those offered by commonly used methods, such as the Nearest Neighbour (NN) strategy or simple Genetic Algorithms (GA) (Konak et al, 2006; Pérez, 2004).

The following section presents a survey of the relevant literature that guides our approach. Section three describes the tuna vessels FAD recovery problem. Section four compares different prediction models in order to show that the final choice (in our case Newton's motion equation) depends on the specific characteristics of each forecasting initiative. Section five shows the experimental design, section six discusses results and, finally, section seven concludes highlighting the main contributions of the paper and their implications.

2.3 Literature review

Calculating the optimal route for recovering N moving elements lies within the Traveling Salesman Problem (TSP) (Gutin and Punnen, 2002). Given a list of cities and their pairwise distances, the task is to find the shortest possible route to visit each city only once and then return home (Applegate et al, 2007). Not surprisingly, the initial applications to real world problems were mainly in transportation and logistics (Dantzig et al, 1954).

Scholars soon perceived, however, that further applications could be feasible if they interchanged the city concept with, for example, soldering points or DNA fragments, and the distance concept with other constraints like traveling times, cost or time windows. Further developments thus appeared in such diverse fields as crystal structure analysis (Bland and Shallcross, 1989), the drilling of printed circuit boards (Grötschel et al, 1991) or even the mapping of a mouse genome (Avner et al, 2001). Certainly, the diverse applications also triggered the development of new problem-solving methods (Tyedmers and Parker, 2012), from exact algorithms to metaheuristics (Blum and Roli, 2003a), such as Swarm Intelligence (Bonabeau et al, 1999) or GAs (Donald, 2010).

GAs represent in fact one of the most consolidated approaches to the TSP (Potvin, 1996). They were first introduced by (Holland, 1975) to generate solutions for optimization problems using techniques inspired by natural evolution (Winter et al, 1996), leading to many theoretical developments over the last thirty years (Reinelt, 1994; Smith and Smith, 2002).

Basically, GAs achieve the optimal solution from a random set of initial solutions called population. Each set comprises an array of numbers where each number represents one of the targets on the route, which are named genes. Hence, each population is evaluated by a fitness measure (in our study, for instance, the measure is determined by the minimal distance between all points on each route), so parents of the next generation are selected probabilistically from the whole population so that the best routes are selected to become the parents of the next generation. The process is regulated by operators reflecting typical gene traits such as *crossover* and *mutation*. GAs repeat this loop until they converge to a near global optimal.

Recently, some scholars have intensified the use of GA to implement theoretical developments in different fields of application such as ship routing with time deadlines (Karlaftis et al, 2009), vehicle routing with time windows (Dumas et al, 1995; Garcia-Najera and Bullinaria, 2011; Wang and Regan, 2002; Baker and Ayechew, 2003) and vehicle routing with loading constraints (Ruan et al, 2013). Despite the progress that implementing GAs brought to the literature on route optimization, however, their potential has not been fully exploited when addressing the TSP with constantly moving targets.

GAs generate near-optimal solutions only when cities are at time $t = 0$; but, in a dynamic scenario, the salesman needs to decide a route for $t = 1$, $t = 2$, etc. The final route that the salesman should follow is therefore necessarily different from the one chosen by a conventional approach to static objectives. This is probably the reason why recent literature has increasingly dealt with dynamic targets, leading to a new line of research in this field since (Psaraftis, 1988) introduced a first reflection on the Dynamic Traveling Salesman Problem (DTSP).

Some contributions compare DTSP and TSP and reflect on basic issues to solve the problem, appropriate approaches, or key evaluation criteria (Huang et al, 2001; Zhou et al, 2003). Most of the literature, however, presents specific applications based on well-known metaheuristics such as Ant Colony Optimization (Eyckelhof and Snoek, 2002; Guntsch et al, 2001), Simulated Annealing (Jeong and Kim, 1991), Tabu Search (Fiechter, 1994) and Genetic Algorithms (Moon et al, 2002; Younes et al, 2003; Liu et al, 2009), under which we can also include particular offshoots like inver-over operators (Li et al, 2006; Yan et al, 2007) or CHC Algorithms (Simões and Costa, 2011). All this work represents a generalization of TSP in which targets are not necessarily static and applications are often formulated with time-dependent variable constraints.

Taking this background into account, our approach resembles that of the existing literature on DTSP but differs in an important way. Both assume the dynamic nature of targets, but the available DTSP solutions work basically on a real time basis to find the changes between nodes (Zhou et al, 2003; Hajjam et al, 2013). The main DTSP methods (Pantrigo and Duarte, 2013) consist in fact in (i) restarting the search method from start, which entails that problems are dealt with as a series of static optimization problems with no relation to each other, and (ii) starting from the best solutions found before the last event, which has found different methodological alternatives. (Garrido and Riff, 2010), for instance, use a hyper heuristic approach generating a set of low-level heuristics; (Pantrigo and Duarte, 2013) rely on a Scatter Search Particle Filter (SSPF) to take advantage of the best solutions obtained in the previous executions; whereas (Hajjam et al, 2013) employ an intermediate structure in a hybrid method that manipulates the self-organizing map in order to minimize route lengths and customers waiting time. Hence, although DTSP has allowed us to gain new insights by addressing new problems, the current literature does not anticipate the future movement of targets; optimal solutions are given only when changes happen and the algorithm is subsequently recalculated. By contrast, our approach follows the line of research drawn by the literature on DTSP, but assumes that changes in the localization of targets are small, traceable and time dependent. This allows us to estimate the trajectories of the targets, as explained above, to subsequently build a genetic algorithm that searches for the best possible route considering their future locations.

2.4 The tuna vessel FAD recovery problem

An FAD is a man-made object used to attract ocean-going pelagic fish, such as tuna, which gathers around it for reasons that are still unclear (Bach et al, 1998). Most purse seiners for tuna therefore release FADs into the sea, letting them float on the water surface following the ocean currents. Some of them are furthermore equipped with echo-sounders that transmit information about their localization and the aggregated biomass beneath them (Castro et al, 2001). Vessels can thus receive new messages from the buoys, and each buoy position is updated at least every 12 hours (latitude and longitude). These messages are transmitted automatically in a predictable and controlled way, communicating in real time via satellite telecommunication systems such as Argos, Inmarsat, Orbcomm and Iridium (Moreno et al, 2007).

Each tuna vessel can handle a different number of FADs, but the maximum per vessel could be some hundreds. Each of them drifts on a particular course and speed, and these conditions change with time because they depend basically on the sea currents under the FAD and on superficial wind. Their speed can thus range from 0.2 knots[1] to 2 knots when they are in areas with strong currents, but on average they travell at about 1 knot.

Most tuna vessels recover their FADs still today following the Nearest Neighbor strategy (NN) (or even without any plan at all). The advantage of this method is that it is easy to implement, as the next target will always be the closest to the vessel. The amount of nautical miles covered by vessels, however, is far from being optimal. In order to calculate the NN with dynamic targets, it is necessary to take the next recovery decision once the previous one has been taken, and once the rest of the targets' movements have been ascertained in order to figure out which will be the closest at time $t+t'$.

Figures 2.1 and 2.2 show a real scenario of FADs drifting on the Indian Ocean. Each black point represents a different buoy, whereas the red line represents the past positions of the object sent via satellite. The time difference between positions is generally 12 hours or less.

2.5 Drifting object prediction

The literature offers different methods to simulate and predict current movements in the sea (Özgökmen et al, 2000), but all are based on complicated mathematical models which require data that a vessel cannot easily obtain, such as wind data and eddy effects. Furthermore, by contrast with Lagrangian buoys, which have scientific purposes, FADs are made without any standard. They are made by fishermen who use simple materials like wood, string and net, so two FADs cannot be assumed to drift identically under the same conditions. It is not possible therefore to use models based on standard Lagrangian buoys to predict the future positions of FADs in the sea.

In this context, we address the drifting FAD prediction problem from another, simpler point of view; one that requires no more data than the last position of each object. As soon as the buoys transmit their subsequent positions, the algorithm will update the last position and will be calculated again to predict the best route, following the current position of the buoy. So if the prediction for a specific FAD is not accurate enough to predict the best route, the solution will be updated when the next message is received, therefore showing a better optimal route if one exists.

[1] 1 knot = 1 nautical mile per hour; 1,852 kilometers per hour

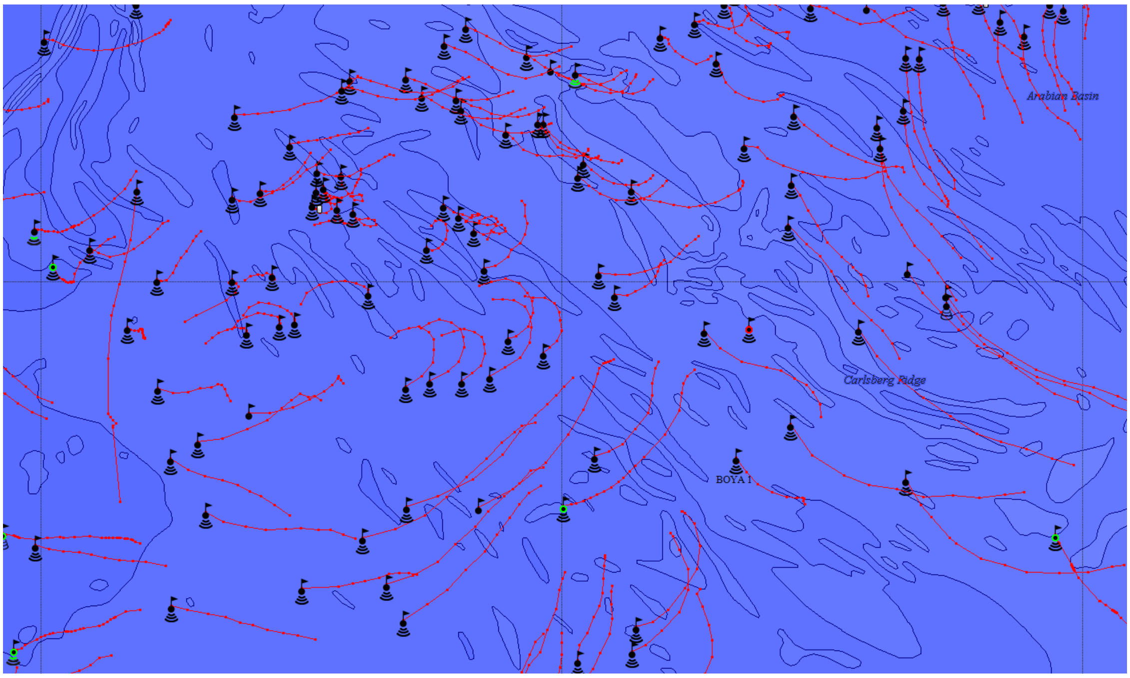

Figure 2.1: FADs drifting in the Indian Ocean

Among the most simple prediction methods available, we have selected to compare two easy-to-implement tools: time series forecasting and Newton's motion equation. These prediction methods are valid for any FAD, regardless of the ocean where it is drifting; however, their effectiveness decreases with time because the error has a cumulative effect. Be that as it may, our goal is to predict where the FAD will be in the near future; not only its next position, but many future positions.

2.5.1 Time series

A time series is a sequence of data points that are measured at uniform time intervals. Time series forecasting, in turn, refers to a model that predicts future events based on past values (Casdagli, 1989). Among the methods to perform this type of analysis, the autoregressive model (AR) is very often used to predict the future position of objects (Reinelt, 1994; Besse et al, 2000). It predicts the output of a system based on its previous outputs.

There are a number of different notations for time-series analysis, among which one of the most common is the following:

$$Y = \{Y_t : t \in T\}$$

This $AR(p)$ notation indicates an autoregressive model of order p (number of lags). The $AR(p)$ model is defined as:

$$y_{t+1} = c + \sum_{i=1}^{p} (\alpha_i y_{t+1-i}) + \varepsilon_t$$

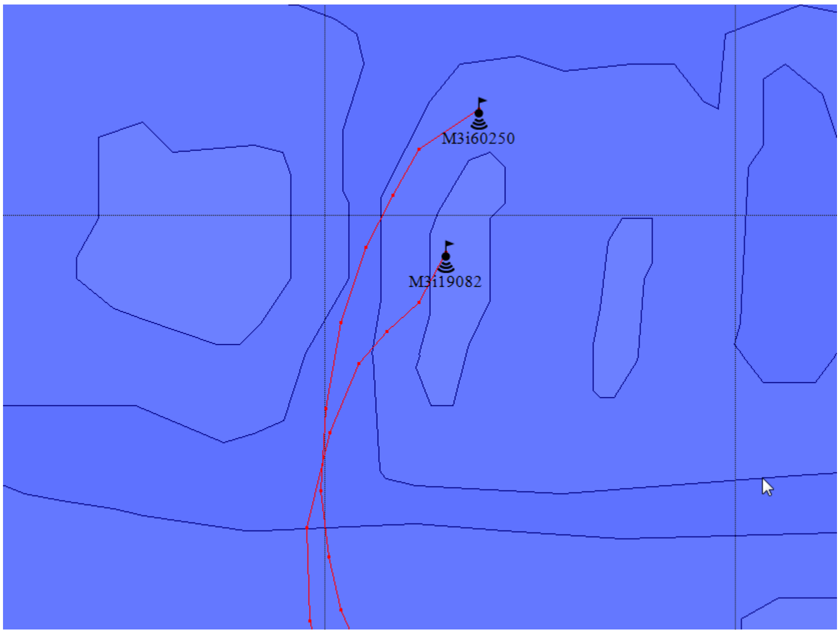

Figure 2.2: Two FADs drifting in the ocean

where $\alpha_1, \ldots, \alpha_p$ are the parameters of the model, c is a constant (often omitted for simplicity) and ε_t is the error. *AR* models are easy to calculate and are widely used as predictors for time series analysis of, for instance, stock markets, etc. (Marcellino et al, 2006; Zhang and Qi, 2005).

2.5.2 Newton's motion equation

Motion equations describe the behavior of a system as a function of time. More specifically, the equations of motion describe the behavior of a physical system as a set of mathematical functions in terms of dynamic variables: normally spatial coordinates and time are used. The Newton's motion equation is considered between two points of time: one initial point and one current or final point:

$$y_{t+1} = y_t + v_t \Delta_t + \frac{1}{2} a_t \Delta_t^2$$

where:

- y_t: the position at the end of the interval (displacement)
- v_t: the velocity at the end of the interval t
- Δ_t: the time interval between the initial and current states
- a_t: acceleration at time t

Notice that in the rest of the article we will refer to the time variable t as discrete. Although FADs are constantly moving in the sea, the information on their position is given twice a day because of the airtime cost inherent to satellite communications.

Given that the acceleration value depends on the speed of the object, in our discrete case its calculation requires the last two positions of the object:

$$a = \frac{\Delta_v}{\Delta_t}$$

2.5.3 Model comparison

Twelve real random FADs were selected from all the oceans to compare how well the prediction methods forecast their next position. Each position is a pair of two single elements: latitude and longitude. Independence between latitude and longitude is assumed so, for a single position prediction, the model estimates two independent parameters, latitude and longitude, which are estimated using the same equation.

The first step is to compare the different time series to see what order suits better the real tracking of FADs. Ordinary Least Squares (OLS) was used to determine AR, and the order of AR was selected using MAPE (Mean Absolute Percentage Error).

MAPE expresses accuracy as a percentage of the actual data and is defined by the formula:

$$M = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{(A_t - F_t)}{A_t}\right|$$

where A_t is the actual value, F_t is the predicted value and N the number of fitted points. MAPE therefore provides an intuitive way to asses the importance or errors, since it easily reflects, for instance, that an error of 10 when the actual value is 100 (10% error) is worse than an error of 10 when the actual value is 1000 (1% error).

We have checked that first order time series predicts much better than other time series with more order lags. The rationale is that the speed of FADs ranges from 0,2 to 2 knots, so the best approach to the next position will be close to the last position, which is what the first order equation suggests.

This first order series is very simple to calculate and only requires the last position of each object, but it has one important limitation. It predicts the future rather accurately once the α_1 parameter value has been set for each FAD. This means that, before predicting the next positions, it is necessary to analyze all the past positions, calculate the best parameter (coefficient) and only then can the equation be used to predict future positions for that FAD (but only for that FAD). If we need to predict future positions of other FADs, the optimal coefficient needs to be re-calculated.

Accordingly, in order to estimate the future movement of FADs with a time series model, we will consider two options: The first stems from estimating the average of all the coefficients, which would be close to 1 ($\alpha \simeq 1$). This is named a random walk model, with the form $y_{t+1} = y_t + \varepsilon_t$, therefore suggesting that the next movement of an FAD is only determined by its last position (which entails there is no movement). The second option estimates the coefficient for each FAD. We will then compare these two time series models with Newton's motion equation.

$$y_{t+1} = y_t \tag{2.1}$$

$$y_{t+1} = \alpha_1 y_t \tag{2.2}$$

$$y_{t+1} = y_t + v_t \Delta_t + \frac{1}{2} a_t \Delta_t^2 \tag{2.3}$$

The information available on each object is latitude and longitude in degrees, and the time difference between each position is 12 hours. Each buoy in the study has a minimum of 200 samples, which is enough to establish the coefficients for each buoy and to compare the different methods properly. Table 2.1 shows some real data from a buoy and how the latitude and longitude change with time. In this case, day 0 refers to the last position, whereas each half-day step is the former position that the buoy sends every 12 hours.

Table 2.1: Buoy data samples

An example of buoys data		
Days	**Latitude**	**Longitude**
0.00	-8.07450	53.08117
0.50	-8.20417	52.92067
1.00	-8.43233	52.75367
1.50	-8.67017	52.56317
2.00	-8.69100	52.45533
2.50	-8.60883	52.27750
3.00	-8.60100	52.08517
3.50	-8.61000	51.87983
4.00	-8.60050	51.78600
4.50	-8.51550	51.74350
5.00	-8.38250	51.67350
5.50	-8.28683	51.54150
6.00	-8.22217	51.45200
6.50	-8.18550	51.41283
7.00	-8.10300	51.42383
7.50	-8.00983	51.43133
8.00	-7.94450	51.38983
8.50	-7.88600	51.34833
9.00	-7.84233	51.33783
9.50	-7.78750	51.24567
10.00	-7.81867	51.19767
10.50	-7.87417	51.12983
11.00	-7.98883	51.09517
11.50	-8.03833	51.08083
12.00	-8.07300	51.07117
12.50	-8.08317	51.01017
13.00	-8.13817	50.92533
13.50	-8.22083	50.84883
14.00	-8.27383	50.85417
14.50	-8.21600	50.81000

The results are shown in Table 3.1: Newton's motion equation has less error in average than the other two methods (even if we compare it with the specific $AR(1)$ for each FAD). This is an important result because FAD positions can be therefore estimated without great computational costs. By contrast, it is worth noting that Newton's motion equation only works for short time predictions because the error increases with time. Forecasting the trajectory of an FAD for more than five days would require a more complex prediction method, probably internalizing chaotic modeling as with ocean currents or weather forecast (Casdagli, 1989). This is not needed here, however, because vessels spend normally less than five days fishing and recover two to three FADs each day. Future research can indeed identify applications where this effort is useful.

Table 2.2: MAPE for three prediction methods

Buoys	Lat/Lon	Ramdom walk	AR(1)	Newton
Buoy 1	Latitude	0,84%	0,84%	0,82%
	Longitude	0,22%	0,18%	0,24%
Buoy 2	Latitude	0,46%	0,47%	0,48%
	Longitude	0,11%	0,11%	0,08%
Buoy 3	Latitude	1,31%	1,28%	0,63%
	Longitude	0,27%	0,29%	0,06%
Buoy 4	Latitude	1,69%	1,67%	0,75%
	Longitude	0,25%	0,32%	0,06%
Buoy 5	Latitude	5,33%	5,37%	1,02%
	Longitude	0,07%	0,14%	0,02%
Buoy 6	Latitude	0,30%	0,33%	0,25%
	Longitude	0,05%	0,13%	0,03%
Buoy 7	Latitude	1,27%	1,28%	0,77%
	Longitude	0,15%	0,23%	0,03%
Buoy 8	Latitude	6,28%	6,30%	3,86%
	Longitude	0,39%	0,44%	0,12%
Buoy 9	Latitude	32,37%	32,36%	18,40%
	Longitude	0,10%	0,16%	0,03%
Buoy 10	Latitude	2,81%	2,79%	0,88%
	Longitude	0,25%	0,27%	0,05%
Buoy 11	Latitude	10,04%	10,04%	5,05%
	Longitude	0,21%	0,26%	0,07%
Buoy 12	Latitude	30,16%	30,16%	59,98%
	Longitude	0,69%	0,72%	0,19%

2.6 Methodology

2.6.1 Data on buoys and vessels

2.6.1.1 Buoys input

If our problem has N drifting objects, being each b_i an FAD:

$$(b_1, b_2, \ldots, b_N)$$

And for each of these objects we know their current position and the last M positions, then the input we have is an $N \times (M+1)$ matrix:

$$\begin{pmatrix} b_1^t & b_1^{t-1} & b_1^{t-2} & \cdots & b_1^{t-M} \\ b_2^t & b_2^{t-1} & b_2^{t-2} & \cdots & b_2^{t-M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_N^t & b_N^{t-1} & b_N^{t-2} & \cdots & b_N^{t-M} \end{pmatrix}$$

Where the first column are all the objects in the current time $t = t$, the second column are all the objects in $t = t-1$, following the progression up to the last column, where we have the objects at $t = t-M$.

Each object b^t has two coordinates, latitude and longitude, to plot the real position on a 2D map:

$$b^t = (latitude^t, longitude^t)$$

Note that M can be different for each object depending on when the object was released in the sea. In our case it is enough to have two positions in the past for each object; that is, $t-1$ and $t-2$:

$$\begin{pmatrix} b_1^t & b_1^{t-1} & b_1^{t-2} \\ b_2^t & b_2^{t-1} & b_2^{t-2} \\ \vdots & \vdots & \vdots \\ b_N^t & b_N^{t-1} & b_N^{t-2} \end{pmatrix}$$

2.6.1.2 Vessel and fishing information input

We need from the vessel the following inputs:

- Vessel speed average (in knots) when traveling from one object to the following one: v_s
- Initial position of the vessel: $v_{init} = (v_{lat}, v_{lon})$
- Time spent by object f_t (fishing time) with two possibilities (our solution can handle both regardless of the choice of the vessel):
  - The same fishing time for all the objects: $f_{t,1} = f_{t,2} = \ldots = f_{t,N}$
  - Different fishing time for each object: $f_{t,1} \neq f_{t,2} \neq \ldots \neq f_{t,N}$

2.6.1.3 Objects prediction

Once we have the inputs, the first step is to predict the next position of each object using Newton's motion equation. Accordingly, the estimation of the R next positions would be:

$$\begin{aligned} \hat{b}_{t+1} &= b_t + v_t\Delta_t + \frac{1}{2}a_t\Delta_t^2 \\ \hat{b}_{t+2} &= \hat{b}_{t+1} + \hat{v}_{t+1}\Delta_t + \frac{1}{2}\hat{a}_{t+1}\Delta_t^2 \\ &\vdots \\ \hat{b}_{t+R} &= \hat{b}_{t+R-1} + \hat{v}_{t+R-1}\Delta_t + \frac{1}{2}\hat{a}_{t+R-1}\Delta_t^2 \end{aligned}$$

The number of future predictions must be enough to mix with the GA. We will talk about a quantity of R future positions, where typically $R >> N$, depending on v_s and f_t.

After predicting R future positions for each object, the result matrix will be:

$$\begin{pmatrix} b_1^t & \hat{b}_1^{t+1} & \hat{b}_1^{t+2} & \cdots & \hat{b}_1^{t+R} \\ b_2^t & \hat{b}_2^{t+1} & \hat{b}_2^{t+2} & \cdots & \hat{b}_2^{t+R} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{b}_N^t & \hat{b}_N^{t+1} & \hat{b}_N^{t+2} & \cdots & \hat{b}_N^{t+R} \end{pmatrix}$$

Where for example $\hat{b}_N^{t+2}$ is the predicted position of the object b_N at time $t+2$.

2.6.2 Genetic Algorithm design

As discussed in Section 2.3, GAs achieve a quasi-optimal solution from a random set of initial solutions called population. In our paper, the specific properties of the GA reflected our concern to minimize the distance traveled by the vessel throughout the recovering process. We summarize them below and provide subsequently in the following subsections a brief report on the algorithm design:

- Population size: 100
- Natural Selection Mechanism: Tournament selection
- Tournament size: 50 couples tournament. Winners are selected
- Crossover type: Greedy crossover. $P = 0,7$
- Mutation type: Simple Mutation between two elements. $P = 0,3$
- Stopping Criteria: 1000 iterations without any fitness improvement

2.6.2.1 Solution encoding

A solution represents the route that a vessel must follow to recover all the buoys. Each buoy will therefore be a point of the route represented by a number. For instance, (4, 2, 3, 1) means that the vessel has to recover object 4, then object 2, object 3 and finally object 1, which is the end of the route; vessels do not return to their initial point of departure.

2.6.2.2 Initialization

As is usual in GAs, the initial population was chosen randomly with the aim of covering the entire search space. We particularly used a random set of 100 initial solutions, which perfectly suits the problem we desire to address (Yang, 1997). The fitness of each solution is measured as the total distance traveled by the vessel to recover all the buoys.

2.6.2.3 Selection

Once the fitness of each random solution has been calculated, the GA works to select a sub-set of routes that becomes the parents of the next generation. We have used here the well-known Tournament Selection Method (TSM) as a selection procedure due to its robustness and simplicity to adjust the genetic pressure, which determines the convergence rate of the GA. Firstly, the TSM chooses a number of couples (tournament size) randomly from the population. We use a 2-Tournament for the mating selection, which entails that we initially selected randomly 50 pairs of routes. Then, each pair competes with each other. The one with the best fitness wins the tournament and becomes a parent for the next generation, named offspring. This selection pressure drives the GA to improve the population fitness through successive generations. In order to do so, nevertheless, the algorithm needs a probability for crossover and mutation.

Following standard practices with GAs, the crossover probability has been set at p=0,7; whereas the mutation probability, p', equals 0,3 (p+p' = 1).

2.6.2.4 Crossover

Crossover is a method where the offspring inherits the characteristics from their parents. We chose the Greedy Crossover Method (Yang, 1997) to address the specific characteristics of our problem: the vessel needs to recover all the buoys only once, and we cannot remove any of them or add others. Thus, given two parent routes R1 and R2, the first offspring is built following these rules: we start in a random buoy b, and then check if the edge leading to b or from b is used in both R1 and R2. If this happens, then the common buoy b is chosen. Otherwise the b's right edge is compared in R1 and R2, so the shorter one is chosen unless it is repeated and it introduces a cycle. In this case, the longer path is chosen. The second offspring is built in a similar way but comparing the b's two left side edges instead of the right ones. In order to implement this method, we need to calculate the distance between some buoys to compare the edges and to determine how the offspring is created. This offspring inherits different characteristics from both parents, ensuring that all the buoys of the route are chosen only once.

2.6.2.5 Mutation

Mutation is used to preserve and introduce the genetic diversity, so it prevents the algorithm to avoid a local minimum when the population is too similar among them. There is always a mutation probability associated to the mutation operator, which as noted above, we fixed at a standard level of 0,3. There are different mutation types; from the simplest where only one chromosome is mutated (bit string mutation) to more complex approximations (Flip bit, Boundary, Gaussian, etc.). Here we use the simplest one, where two elements of the route are exchanged randomly, since it is sufficient to maintain the genetic diversity of our population and ensures proper convergence of the algorithm. For example, if we apply mutation in the route R1= (2, 4, 1, 3, 5) over the elements 2 and 3 (first and fourth position of the vector), the resultant offspring will be R1'= (3, 4, 1, 2, 5).

2.6.2.6 Stopping Criteria

So once we have chosen the 50 parents from the initial population, we provoke crossover or mutation. In both cases we will generate two descendants from each of the 50 parents. If the crossover operator is selected, we choose randomly other parent from the other remaining 49 parents, and later apply the Greedy Crossover technique in order to get the two descendants. By contrast, if the mutation operator is selected, we will mutate two genes (buoys) of the parent route. So if we desire to generate two descendants, we need to perform the mutation twice for each parent. Once we repeat this process for all 50 parents, the offspring will double to reach again 100 –improved– solutions. This process finishes when the loop of steps achieves 1000 iterations without any fitness improvement (stopping criteria). The quasi-optimal route is thus obtained reflecting the shortest distance to recover all the buoys from the vessel's initial position.

2.6.3 GATP final solution: implementing the GA with Trajectory Prediction

Based on the inputs and techniques we have showed previously, this subsection describes how the GA can work together with the prediction technique (in this case Newton's motion equation) to improve the route when targets are constantly moving. The solution, named GATP (Genetic Algorithm based in Trajectory Prediction), will evolve from scratch to a route where the vessel anticipates the future movement of the FADs.

In order to calculate the final route, we will use the predicted positions. Figure 2.3 shows the block diagram of our GATP solution, whereas figure 2.4 represents in detail how our method calculates the fitness of each route.

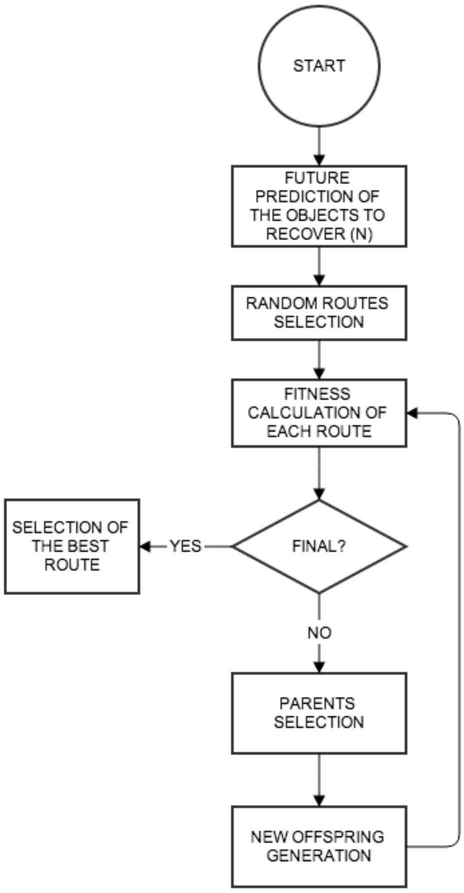

Figure 2.3: GATP Block Diagram

We show below the steps to solve the problem (figure 2.3):

1. Calculation of the R future positions of each object: $(\hat{b}^{t+1}, \hat{b}^{t+2}, \ldots, \hat{b}^{t+R})$.

2. Random solutions are calculated as follows:
   Being $(b_1, b_2, \ldots, b_N)$ the objects to recover, we will select random solutions to have the first generation of solutions to the problem. Each route is a sorted list

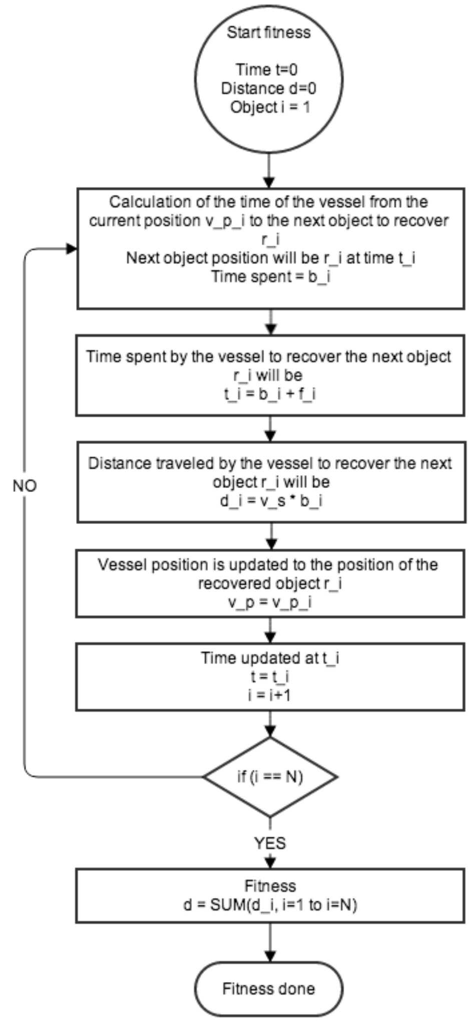

Figure 2.4: GATP Fitness calculation

of the objects: $(r_1, r_2, \ldots, r_N)$, where each r can be whatever object to recover $(b_1, \ldots, b_N)$.

3. Calculation of the fitness of each route:

   (a) $[t = 0]$ and $[d = 0]$. Time and distance equal to zero.

   (b) v_p = initial vessel position.

   (c) From $i = 1$ to N

      i. Calculation of the time needed by the vessel to go from the current position (v_p) to the next object (r_i: next object in the selected route). This time it will be stored as b_i. Also, the fishing time of this object will be taken into account as f_i.
      The position of the object r_i will be $\hat{r}_i^t \equiv$ predicted position of that object i at time t.

ii. Calculation of the time spent to recover and fish in the object r_i

$$t_i = b_i + f_i$$

iii. Vessel position (v_p) is updated to the position of the last recovered object r_i at time t.

iv. Calculation of the distance traveled by the vessel to recover the object i

$$d_i = v_s \cdot b_i$$

v. Current time t is updated to $t = \sum_{j=1}^{i} t_j \equiv$ current time spent.

vi. Total distance traveled:

$$d = \sum_{j=1}^{i} d_j$$

(d) Fitness of the route:

$$d = \sum_{i=1}^{N} d_i$$

4. Check if the termination condition has been reached. In this case the result is the best route; if not, go to step 5.

5. Parents selection.

6. New offspring generation (crossover and mutation).

7. Go to step 3.

Note that we ignore the movement of the FAD in the first movement of the vessel, which means that we ignore the movement of the FAD when the vessel is traveling to recover it. This only happens, however, for the first object. The rationale of this mathematical simplification is to avoid the calculation of the collision vector from the vessel to the object when it is moving (alternatively, the only challenge has to do with the time calculation of the algorithm). This evolving process makes the routes selected in each generation converge on the route that minimizes the real distance from the vessel to all the FADs. Algorithm 1 shows the pseudocode of the GATP solution.

Figure 2.5 shows the rationale of the GATP solution and how the route is calculated from the initial position of the vessel (represented by a square) to each object, considering each trajectory is time dependent. We can see that the first vector goes directly where the first buoy is; however, the second finishes where the buoy is expected to be at time t_{i+1}. The same procedure holds for the rest of the buoys.

The most significant difference between our GATP method and the GA-TSP approach is the restriction used to calculate the costs of traveling from one point to the other. The GA-TSP method does not use any prediction technique and it is based in the static assumption of the objects. For this reason the fitness function is different in each case. The costs are totally dependent on the distance traveled by the vessel before measuring where the next object will be at time t. The distance between the vessel (previous object at time t) and the next object at time t is subsequently calculated. This

initialization;
while $i < N$ **do**
    calculates future positions of $object_i$;
    $i = i + 1$;
end
random routes selection;
i = 0;
for $i \leftarrow 1$ **to** N **do**
    calculation of time from current vessel position to $object_i \rightarrow b_i$;
    calculation of time spent by the vessel to recover the next $object_i \rightarrow t_i$;
    calculation of distance traveled by the vessel to recover next $object_i \rightarrow d_i$;
    update vessel position: $v_p = v_{p_i}$;
    update time spent: t = $\sum_{j=1}^{i} t_j$;
    $i = i + 1$;
end
fitness calculation: d = $\sum_{i=1}^{N} d_i$;
if *stop condition* **then**
    final route = best fitness route calculated;
    exit;
else
    parents selection from routes;
    new offspring generation;
    $i = 0$;
    go back to the for section;
end

Algorithm 1: GATP algorithm

fitness function makes our GATP solution evolve towards the route that minimizes the distance traveled by a vessel.

Figure 2.6 shows our solution with an example of the implementation of the three methods.

Each route has a different color when they deviate from the rest:

- NN: grey color
- GA-TSP: pink color
- GATP: orange color

The FADs are represented by the dots in red. In this example the orange cross represents the starting position of the vessel, whereas each circle reflects an FAD recovery. The final point of each route (last FAD recovered) is marked with a cross inside a circle.

The graphic shows that the circles are close for the first FADs recovered, but as the time goes by, the distance increases. Particularly, the three methods start similarly: all start picking the two FADs to the north of the orange cross (vessel initial position). This is why we can only see one trajectory in orange. Then, the NN solution leads the vessel to a different route; we can see how the grey line goes south, whereas the orange

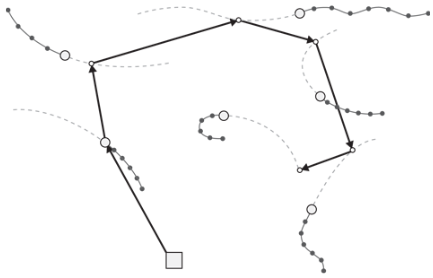

Figure 2.5: Using GA based combining prediction methods

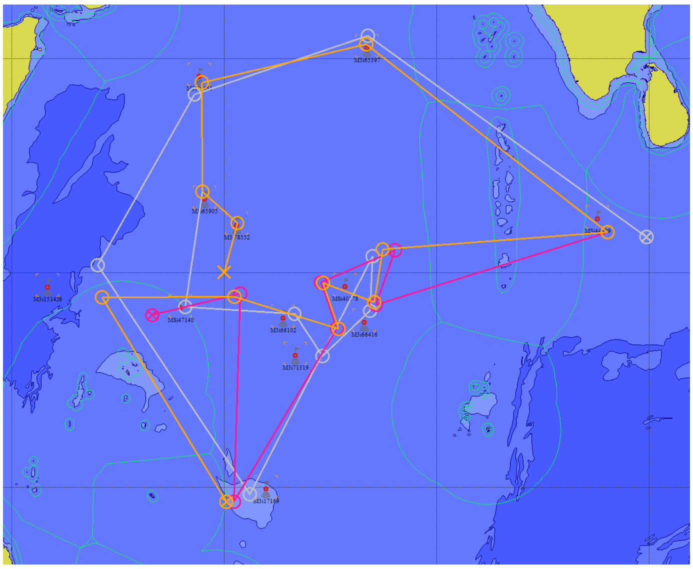

Figure 2.6: NN, GA-TSP and GATP: A graphic example for 12 FADs

line keeps heading north. The GA-TSP and GATP trajectories thus continue the route in the same order during the recovery of 3 more FADs; one to the north, the following to the east and the third to the south-east. After picking the fifth FAD, however, the routes deviate and the pink trajectory reflecting GA-TSP appears.

2.7 Results and discussion

In this section we discuss the improvement achieved by addressing DTSP with the method proposed in this paper: GAs based on Trajectory Prediction (GATP). Initially, and just with informative purposes, we compare the Nearest Neighbor (NN) strategy, which is the method normally used by tuna vessels, with TSP solved by GAs (GA-TSP). This second method consists on applying a simple GA to the TSP problem. Then, we compare the performance of our GATP method with both NN and GA-TSP. It is worth recalling that we use real data offered by tuna companies for the last quarter of 2013. We have made the following assumptions:

- Average vessel speed = 12 knots

- Recovery time = 3 hours for all the objects
- Number of objects to recover = 6, 9 and 12
- Buoys have different speed, which can go from 0.2 knots up to 2 knots
- Distance between buoys is also variable, it can go from 100 nm up to 1,500 nm

The results (average, standard deviation and the improvement percentage achieved between each two methods) are shown in Table 2.3. We can observe that our GATP method is always better than the NN and GA-TSP for recovering 6, 9 and 12 buoys (normal working range for these vessels).

These results are supported statistically. The comparisons have been tested through a Repeated Measures ANOVA, given that the same subjects (vessels) are used for each treatment (method). We thus find significant differences among methods and among the interaction of methods with a different number of FADs (Sig. $= 0.000 < 0.05$) (Table 2.4). Results are consistent since the most frequent multivariate tests used in ANOVA (Phillai's trace, Wilks' Lambda, etc.) show a very high significance. If we now deepen into which of the means for the three methods are significantly different from the others, the pairwise comparisons support our descriptive analysis: average results by GATP are statistically different (distances are lower) from those obtained by NN and GA-TSP for 6, 9 and 12 FADs (Table 2.5).

Table 2.3: Results comparison

			Total distance traveled (nautical miles)			Improvement Comparison		
Experiment	**N^o buoys**		**NN**	**GA-TSP**	**GATP**	**GA-TSP vs NN**	**GATP vs NN**	**GATP vs GA-TSP**
1	6 buoys	$\bar{x}$	1735.1	1645.0	1615.6	4.4%	6.2%	1.9%
		σ	566.6	498.8	501.2	5.6%	5.6%	2.2%
2	9 buoys	$\bar{x}$	4277.0	4185.4	3953.6	1.7%	7.3%	5.5%
		σ	807.3	726.8	702.0	7.5%	4.7%	3.5%
3	12 buoys	$\bar{x}$	4069.4	3817.0	3734.1	5.6%	7.5%	2.1%
		σ	725.8	559.6	556.3	5.6%	8.3%	4.4%

Table 2.4: Repeated Measures ANOVA: multivariate tests

Effect		**Value**	**F**	**Hypothesis df**	**Error df**	**Sig.**
Method	Pillaiís Trace	0.552	34.470	2.000	56.000	0.000
	Wilksí Lambda	0.448	34.470	2.000	56.000	0.000
	Hotellingís Trace	1.231	34.470	2.000	56.000	0.000
	Royís Largest Root	1.231	34.470	2.000	56.000	0.000
Method · N^o of Buoys	Pillaiís Trace	0.371	6.482	4.000	114.000	0.000
	Wilksí Lambda	0.653	6.647	4.000	112.000	0.000
	Hotellingís Trace	0.495	6.805	4.000	110.000	0.000
	Royís Largest Root	0.405	11.556	2.000	57.000	0.000

In short, GATP yields better results than other common optimizing strategies when addressing routes for moving targets in the short term. The sophistication of the prediction method, however, must be adapted to the specific characteristics of the exercise. For instance, improving forecasting accuracy for FADs trajectories in the long term requires a prediction method that considers the chaotic nature of its currents and internalizes Eddy effects, temperature or altimetry. As mentioned above, however, this

Table 2.5: Repeated Measures ANOVA: Pairwise Comparison

Pairwise Comparisons						
					95% Confidence Interval for Differences	
(I) Method	**(J) Method**	**Mean Difference (I-J)**	**Std. Error**	**Sig.**	**Lower Bound**	**Upper Bound**
NN	GA-TSP	144.711	32.937	0.000	63.467	225.956
	GATP	259.415	35.151	0.000	172.709	346.121
GA-TSP	NN	-144.711	32.937	0.000	-225.956	-63.467
	GATP	114.703	17.826	0.000	70.733	158.674
GATP	NN	-259.415	35.151	0.000	-346.121	-172.709
	GA-TSP	-114.703	17.826	0.000	-158.674	-70.733

is not a concern in the specific case of tuna fishing because a typical vessel spends normally less than five days fishing and recovers two sometimes 3- buoys each day.

Finally, although the execution time of the GATP algorithm is an important variable indeed, tuna vessels do not require a real time computation because it takes at least three hours (normally between 8 and 12 hours) to fish and recover each buoy. The skipper will therefore run the solution to plan a week of work, finding out which is the best buoy to start with and then run the algorithm with information updates on the buoys position to keep on. Even if users had to wait several minutes for the algorithm execution in each buoy, it would not accordingly represent a problem. Our experiments with real data show, anyhow, that the execution time for the 3 methods is much lower:

- NN: 10 - 20 milliseconds for the recovery of 6 up to 12 buoys
- GA: 0,5 - 1,2 seconds for the recovery of 6 up to 12 buoys
- GATP: 0,6 - 2 seconds for the recovery of 6 up to 12 buoys

2.8 Conclusions

We have addressed the Traveling Salesman Problem with GA assuming that targets change their position with time. Our contribution is a new way of solving the dynamic route optimization problem using a simple prediction method that, combined with the power of GAs, makes the implemented algorithm evolve towards the near-optimal route.

The comparative analysis between GATP and other commonly used methods like NN or GA-TSP reveals the benefits of internalizing predictive methods within GAs. However, given the chaotic nature of ocean currents and regardless of the sophistication of the forecasting method, we can expect that GATP adds less value if long term predictions were needed.

In practical terms, the GATP algorithm's execution time allows new, better routes to be recalculated easily when the FADs new positions are updated, also showing a better real time route possibility where it exists. We could accordingly conclude that GATP allows tuna vessels and any other agent pursuing moving targets in the short term -like military airplanes- to minimize the distance traveled, which would impact directly on such relevant variables as the time employed, fuel consumption or CO_2 emissions to the atmosphere. It is important to emphasize here that, for a given speed, the distance saved is equivalent to fuel savings. This is extremely relevant not only to reduce costs but also to increase the storage space.

Furthermore, the development of more sustainable fishing with FADs may benefit from further research. To begin with, the algorithm is totally flexible and open to future improvements adding new restrictions, such as prioritizing the recovery of some targets that have more fish beneath them (using buoys with echo-sounder information), working with time-windows for the recovery of FADs (the vessels can't get fish during the night, for example), implementing a multiple vessel FAD recovery strategy or finding the optimal vessel speed in order to save more fuel. From a more general perspective and beyond the specific tools employed in this paper, our results also reflect the value of mixing an heuristic method with a predictive technique, regardless of the specific choice in any of the two. A quasi-optimal solution could be consequently found using other heuristic methods if they were combined with prediction techniques in a proper way.

Finally, from a theoretical point of view, it is worth noting that GATP is a generalization of the classic methods to solve the TSP with GA. When targets are not moving the predicted next position by GATP will be the same offered by GA-TSP, so this solution will evolve as classic methods for GA-TSP do. However, when targets start moving, the proposed solution is different because GATP evolves in order to continue optimizing the total route traveled, assuming the future movement of each object and therefore achieving better results (depending on the prediction period). Our solution can therefore be used in a generic way; for static, dynamic and mixed scenarios, being a more flexible and more adaptable solution to estimate near-optimal routes in general.

Acknowledgements

The authors wish to thank Marine Instruments for the buoys data. We also thank Iago Paz, Gabriel Rosón, Marcos Álvarez, Diego Piñeiro, Francisco Pino, Bruno Lema and Daniel Mosquera for several useful suggestions. Regarding financial support, this research benefited from the "Programme for the Consolidation and Structuring of Competitive Research Units" co-funded by the European Regional Development Fund (10SEC300036PR), as well as from the support of the Spanish Ministry of Economics and Competitiveness through project ECO2013-45706-R.

3

Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels

En este capítulo se estudia el problema de optimización del problema del viajero múltiple en entornos dinámicos (MDTSP). Este problema es la ampliación natural del primero de los trabajos presentados (DTSP), pero para el caso general cuando el número de viajeros o recolectores es mayor que uno.

El MDTSP es muy común dentro de los problemas de optimización logística, pero es particularizado para el caso concreto de los barcos atuneros que usan objetos a la deriva para la pesca (FADs), pues en muchas ocasiones estos barcos trabajan en grupo compartiendo los objetos, siendo el problema de optimización al que se enfrenta el grupo de barcos un claro ejemplo del MDTSP.

El MDTSP presenta una gran complejidad computacional a la hora de encontrar una solución al problema. Al igual que el DTSP, se trata de un problema NP completo, pero con todavía más variantes debido al incremento del número de viajeros. Este problema también se encuentra dentro de la literatura como VRP (Vehicle Routing Problem), siendo la diferencia fundamental con el MTSP el uso de restricciones al problema, algo que convierte al VRP en una generalización del MTSP, con aplicaciones particulares en transporte y logística. Las restricciones pueden ser la distancia máxima recorrida por los agentes, la capacidad de cada uno de ellos u otros.

El trabajo realizado usa la literatura del MTSP como base, pero la solución obtenida es también válida para el entorno del VRP.

La forma de abordar la solución al MDTSP se realiza diseñando un algoritmo genético que resuelva el MTSP, combinándolo con técnicas predictivas.

El algoritmo diseñado usa como parámetros las posiciones de los objetos (FADs), desde las más actuales a las pasadas. Además también usa la velocidad a la que el barco se va a desplazar y el tiempo que va a estar pescando en cada objeto. A través de estas variables, combinando la predicción con un algoritmo genético, el algoritmo evoluciona hacia una solución que se adapta al escenario dinámico en el que se encuentran estos objetos.

Los resultados obtenidos son contrastados con datos reales, observando que el algoritmo diseñado muestra importantes mejoras en términos de eficiencia y, por lo tanto, en el ahorro de costes al obtener mejores rutas a la hora de recoger los objetos a la deriva.

De manera indirecta, pero no menos importante, la solución propuesta permite que se reduzcan las emisiones de CO_2 a la atmósfera de manera significativa.

Al igual que sucede en el trabajo presentado anteriormente, la solución propuesta resulta ser una generalización válida para los problemas MTSP en entornos dinámicos y estáticos, con aplicación directa en el mundo de la optimización logística como se muestra aquí, pero también aplicable a otras disciplinas.

Este trabajo ha motivado una publicación en el journal *'Engineering Applications of Artificial Intelligence'* en el año 2018, en coautoría de Antonio Sartal y Xosé H. Vázquez.

3.1 Abstract

The multiple Traveling Salesman Problem (mTSP) is a widespread phenomenon in real-life scenarios, and in fact it has been addressed from multiple perspectives in recent decades. However, mTSP in dynamic circumstances entails a greater complexity that recent approaches are still trying to grasp. Beyond time windows, capacity and other parameters that characterize the dynamics of each scenario, moving targets is one of the underdeveloped issues in the field of mTSP. The approach of this paper harnesses a simple prediction method to prove that integrating forecasting within a metaheuristic evolutionary-based method, such as genetic algorithms, can yield better results in a dynamic scenario than their simple non-predictive version. Real data is used from the retrieval of Fish Aggregating Devices (FADs) by tuna vessels in the Indian Ocean. Based on historical data registered by the GPS system of the buoys attached to the devices, their trajectory is firstly forecast to feed subsequently the functioning of a genetic algorithm that searches for the optimal route of tuna vessels in terms of total distance traveled. Thus, although valid for static cases and for the Vehicle Routing Problem (VRP), the main contribution of this method over existing literature lies in its application as a global search method to solve the multiple TSP with moving targets in many dynamic real-life optimization problems.

3.2 Introduction

This paper addresses the synergies in combining a predictive technique with a metaheuristic evolutionary-based method to solve the multiple Traveling Salesman Problem (mTSP) with moving targets (mTSP-MT). The mTSP-MT is the generalization of the well-known Traveling Salesman Problem (TSP). It deals with multiple salesmen, and targets (e.g., customers or objects) are not fixed. As in any TSP, however, the aim is to minimize the total distance traveled by all salesmen.

The mTSP-MT is therefore more suitable than the ordinary TSP for a wider range of real-world problems. In fact, this is the method used, for example, in the defense sector to protect an airport or a security zone from mobile intruders (raiders, animals, vehicles, etc.), or in the logistics sector to supply a fleet of boats or mobile ground units (Stieber et al, 2015; Stieber and Fügenschuh, 2017). It can also be applied to the Vehicle Routing Problem (VRP) with multiple vehicles, time windows or capacity restrictions (Sundar et al, 2017; Bae and Chung, 2017). New potential uses are also emerging every day in mobility and delivery services (e.g., delivery services, real-time mobility requirements, drones scheduling and collaboration, etc.) conducted by companies such as Uber and Amazon (Menezes et al, 2015).

Whereas the diverse perspectives and problem-solving methods have helped practitioners and scholars to address a multitude of TSPs, including mTSPs in various industries (Menezes et al, 2015), the literature on mTSP-MT is still scarce. This is possibly due to the greater complexity of this type of problems compared to conventional TSP approximations, which may also explain why ad hoc experiments with many restrictions (small distances, planned routes, fixed starting-points, etc.) have often ended with few or not applicable solutions for real-life problems. For example, (Liu, 2013) and (Jiang et al, 2005) proposed various solutions for the mTSP-MT problem by narrowing the scope of analysis to one dimension and limiting the working speed. Similarly, other studies have restricted the positions of the salesmen (e.g., by having them start at the same point, located in the middle of the area) or the possible targets' movements (e.g., by forcing customers to move in a structured path) (Menezes et al, 2015; Stieber and Fügenschuh, 2017). In addition, regarding the calculation methodology, the most recent approaches have worked on a real-time basis and have been recalculated to find the changes between nodes (Zhou et al, 2003; Hajjam et al, 2013). However, they have not anticipated the targets' future movement, so the optimal solutions appear only when the changes are communicated and the algorithms have been recalculated.

Against this background, the paper addresses the problem of moving targets by combining a predictive technique (Newton's movement equation) with a genetic algorithm (GA). This new approach, which could be named genetic algorithm based in multiple-trajectory prediction (GAMTP), yields a generic solution that not only suits dynamic and static scenarios, but it also applicable to any real-world problem with multiple travelers and mobile targets. GAMTP thus combines prediction and GA in a single method to reach a better global optimization solution than when GAs alone used without internalizing prediction.

Since the objective of this work is to prove that integrating forecasting within a metaheuristic method (e.g., genetic algorithms) better results are achieved than in the simple non-predictive version; we chose Newton's movement equation as our predictive technique over other techniques because it offers a quick, short-term prediction even when provided with very little information (Groba et al, 2015). Analogously, GAs were chosen as metaheuristic method for three reasons: (1) they are evolutionary, which is mandatory for the algorithm implementation; (2) they show fundamental properties in terms of robustness and statistical convergence; and (3) they can reach a solution within an acceptable computational time (Jih and Hsu, 2004).

Data come from different group of tuna vessels retrieving their fish aggregating devices (FADs) in the Indian Ocean during April in 2017. FADs drift in the sea and provide an artificial substrate for attaching organism such as algae and invertebrates. This phenomenon probably stimulates a food chain that attracts different type of fish. Tuna also tends to gather beneath them. All FADs are attached to a buoy with a Global Positioning System (GPS) that transmits its coordinates every 12 hours. The people steering the vessels, which work in groups, need to design their routes to recover the constantly moving FADs in order to minimize the total distance traveled. We compare our results with both the most commonly used method, the nearest neighbor (NN) strategy, and a classic mTSP approach based on GA (i.e., without prediction) (Bjarnadóttir, 2004).

The paper is organized as follows. The next section provides a review of the literature. Sections 3.4 and 3.5 describe, respectively, the data and methodology. Section 3.6 introduces the model and presents the experimental design. Section 3.7 discusses results and, finally, Section 3.8 concludes by highlighting the paper's main contributions and its implications.

3.3 Literature Review

TSP is a well-known of Combinatorial Optimization (CO) problems that is NP-hard (Garey and Johnson, 1983), and in which, assuming that $P \neq NP$, no polynomial time algorithm exists (Karp, 1972).

An extension of TSP involves more than one salesman (mTSP), and assumes that each city must be visited exactly once and by only one salesman (Bektas, 2006a; Venkatesh and Singh, 2015). Thus, given a start-and-end point (a depot), a set of n cities to be visited by one salesman, and m salesmen (where $n > m$ the optimal), then the mTSP consists of finding routes for all m salesmen such that the cost of visiting all cities is minimized. The cost can be defined in terms of distance, time, or other criteria. Thus, although mTSP is NP-hard like TSP but entails a more complicated problem because cities must be assigned firstly to each salesman, and then the optimum order is subsequently determined for each salesman. Two main versions of the mTSP can be defined based on the number of depots. In the first version all m salesmen start and end at one depot. In the second version every salesman begins and ends at a different depot. We address the second variant, which represents a more generalized situation that aims at minimizing the total distance the salesmen travel (i.e., the total length of all routes). This approach reflects the strand of literature dealing with multiple Traveling Salesman Problem (Bektas, 2006b).

Furthermore, the solution shows an additional trait: targets are not fixed and can vary their positions over time. This variant is known as mTSP with moving targets (mTSP-MT), which is a dynamic generalization of the mTSP that makes the problem more suitable to a wider range of real-world situations in various industries. In fact, many mTSP-MT applications exist, for example, in supply logistics (Stieber et al, 2015), robotic patrolling (Pushkarini Agharkar and Bullo, 2015), scheduling and routing (e.g., bank-crew scheduling, workload balancing, and school-bus routing), as well as in the defense sector (e.g., the multiple-weapons-to-multiple-targets assignment problem) (Stieber and Fügenschuh, 2017). On a broad perspective, one of the fields that could highly benefit from this type of analysis in the future is the transportation and delivery service (including unmanned aerial vehicle services) leaded by companies such as Uber, Amazon and others (Agatz et al, 2016; Dorling et al, 2017).

Curiously enough, however, there are relatively few approaches to solve mTSP-MTs. Again, this is possibly due to the greater complexity inherent to the dynamics of mTSP-MT in comparison with traditional TSP (Garcia-Najera and Bullinaria, 2011; Hajjam et al, 2013) or the simpler moving-target TSP (e.g. a supply ship that resupplies patrol boats as they work, a fishing boat collecting its catch at sea, or an airplane that must intercept a number of mobile ground units) (Helbing and Tilch, 1998; Groba et al, 2015). There are also variants of the TSP-MT (including one with resupply) in which the salesmen must return to the depot after intercepting each target (Jiang et al, 2005; Liu et al, 2009; Jindal and Kumar, 2011).

Whereas this specific literature has helped researchers to explore the TSP-MT, it has proceeded so far with a high number of restrictions in order to reach a feasible result (Helvig et al, 2003; Blum and Roli, 2003a). The cost of this strategy, nevertheless, is that they often end up with few or not applicable real-world solutions. For instance, (Jiang et al, 2005) described a solution approach based on GA with a fixed number of cities in which the target moved at a constant velocity, which in fact is a very common assumption in this area. The same speed restriction has been considered in moving-target TSP situations (Helbing and Tilch, 1998; Helvig et al, 2003). Similarly, other studies have restricted the salesmen's position (e.g., by requiring that they start at the same point in the middle of the area) or the possible targets' movements (e.g., by having the customers only move in structured paths) (Menezes et al, 2006; Stieber et al, 2015). In the same way, (Jindal and Kumar, 2011) assumed all targets started from their starting position, were only in one dimension, and moved with constant velocity. In general terms, therefore, this literature shows that the available research can hardly solve practical problems; rather, it is mainly focused on providing structure and analyzing variants of the TSP that answer specific questions in made-to-measure approximations of reality.

Similar arguments hold for VRP (Braekers et al, 2016; Toth and Vigo, 2014), which could be considered as a generalization of mTSP with particular applications to transport and logistic (Montoya-Torres et al, 2015). The main difference between the classic moving TSP and the moving VRP is that the VRP can include additional restrictions beyond distance, such as added vehicle capacity, time constraint, a known non-negative demand for each depot, and a non-negative cost for each route (Eksioglu et al, 2009). Nevertheless, just as it happens with mTSP, research on VRP with moving targets is still underdeveloped. The existing literature focuses on Unmanned Aerial Vehicles (UAVs) (including combat UAVs), surveillance missions, and military needs, but none of these studies offers a generic solution with no restrictions (Shetty et al, 2008; Geng et al, 2014; Shima and Schumacher, 2005). Thus, although our research hinges on mTSP-MT, it can also contribute to solve VRP problems (Cattaruzza et al, 2017).

Summing up, there is a gap in the literature on mTSP and VRP with regard to dynamic scenarios. Research so far has worked with basic settings and simplified parameters that not only lead to a continuous recalculation of the solution, but make this very same solution of limited application to real-life situations. Our proposal, however, addresses simultaneously three key issues that characterize any real-world situation: (1) multiple targets for (2) multiple salesmen and (3) in dynamic scenarios. This makes it a generalized solution for static and dynamic scenarios of mTSP and VRP, and opens multiple real-world applications in many scientific and business fields, from medicine or physics to production and logistics.

3.4 Data

3.4.1 Introduction: FAD recovery for tuna vessels

The global tuna fishery is one of the largest in the world. Aggregate catches of tuna and associated species, including swordfish and other billfishes, reached a record level of 6.6 million tons in 2010 (Food and Agriculture Organization of the United nations, FAO, 2012). The most widely used and fastest-growing fishing gear for targeting tuna is the purse seine. Since the 1950s, purse-seining vessels have benefited from the adoption of power blocks, increases in fish-holding capacity and freezing technology, improvements in tuna-locating techniques (e.g., helicopters, bird sonar, GPS, and most recently, UAVs), and the use of FADs (Miyake et al, 2010).

FADs are human-made structures that facilitate the attraction and aggregation of ocean-going pelagic fish such as tuna (Rajeswari, 2009), so fishermen have traditionally seeded them throughout the oceans to make their job more efficient. Still today, most FADs are handcrafted. This is a relevant issue because this specific trait make FADs drift on diverse courses and at different speeds. Furthermore, courses and speeds change with time because they depend on the sea currents and on superficial wind. For instance, currents beneath FADs move in a range from 0.2 knots[1] to 2 knots.

Each FAD is tied to a satellite buoy that sends information to the seiners from the FAD's on GPS position, battery level and water temperature. Simultaneously, most of the buoys are also equipped with echo sounders that transmit information about the aggregated biomass beneath them (Castro et al, 2001). Vessels receive messages on all these parameters from the buoys at least twice a day via real-time satellite communication systems, such as Argos, Inmarsat, Orbcomm or Iridium (Moreno et al, 2007). Seiners can thus track FADs and the biomass beneath them in real time.

Information and communication technologies have thus expanded significantly the number of purse seiners using FADs during the last 15 years (Hallier and Gaertner, 2008). Some estimates suggest there are around 120,000 FADs deployed in the oceans. It is not surprising, consequently, that the use of FADs has been restricted in recent years to control fishing activity and to maintain better stocks of fishing resources. Organizations such as the Indian Ocean Tuna Commission apply normative rules to limit the quantity of FADs that each vessel can handle. For this and other reasons, each tuna vessel may share its FADs information among groups of two to five vessels, depending on the size of the company. A single vessel, on the other hand, can handle as many as hundreds of FADs.

Once the basic organization of the industry has been presented, it is worth noting that fuel consumption has been shown to be a major contributor to the operating costs of tuna fishing vessels, typically representing between 30 and 75% of total operating costs (Miyake et al, 2010). Vessels that fish tuna with purse seine have an average fuel-use intensity, weighted by landings, of 368 L/t (Parker et al, 2015). Given a moment t, therefore, the challenge for tuna vessels is to figure out the best route for each vessel to maximize fishing and minimize travel. There are no restrictions regarding where the vessels are at time t or where they have to finish their tours. Under these circumstances, the majority of tuna vessels follow the Nearest Neighbour (NN) strategy, which is basically easy to implement. However, when the number of FADs to recover is greater than one, the constant movement of FADs make the NN method less useful.

Figure 3.1 represents the problem that the vessels face. Many FADs are drifting in the ocean, and some vessels that share FAD information need to plan their recovery. The FAD current position is represented in black; the white part reflects the drift in the last days. We can also observe the current positions of three vessels (red, yellow and green symbols) at time t. Hence, the figure presents the difficult problem that tuna vessels face when designing a common recollection strategy. If the high fuel consumption of tuna vessels is considered, the practical side of the challenge becomes probably more clear.

The mathematical notations used in the article in Table 3.1 can help readers understand the proposed solution.

[1]The knot is a unit of speed equal to one nautical mile per hour, exactly 1.852 km/h

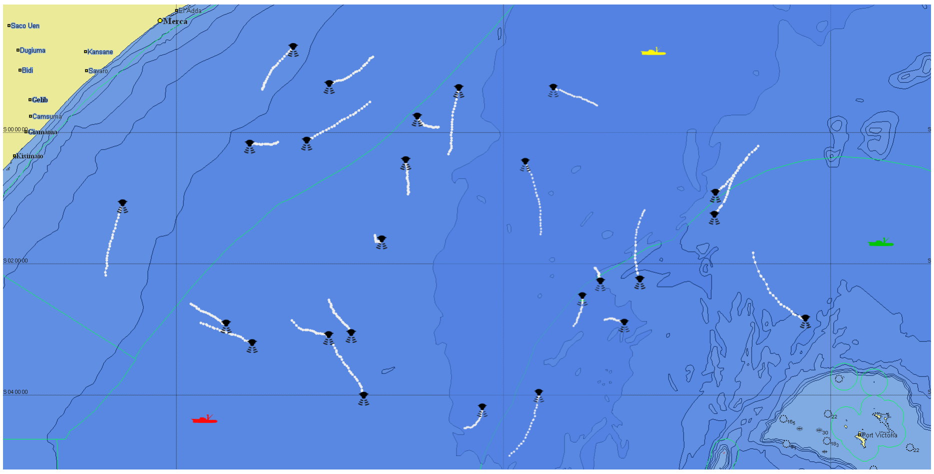

Figure 3.1: Drifting FADs and three vessels in the Indian Ocean.

Table 3.1: Mathematical symbols used.

Symbol	Description
n	number of cities or FADs
m	number of salesmen or vessels
t	time
r	number of predictions. Evolution of time t
f_i^t	position of FAD i at time t
$\hat{f}_i^r$	estimated position of FAD i at time r
v_i	initial position of vessel i
s_i	speed of vessel i
z_i	number of objects (FADs) that salesman (vessel) i has to recover
z_0	index of the initial FAD that a vessel has to pick up
z_f	index of the final FAD that a vessel has to pick up

3.4.2 Data on buoys and vessels

3.4.2.1 FADs input

Our scenario is composed of n drifting objects (FADs), which are labeled f_i: $(f_1, f_2, \ldots, f_n)$. Since we know their last transmitted coordinates as well as their last h positions, the input is an $n \times (h+1)$ matrix:

$$\begin{pmatrix} f_1^t & f_1^{t-1} & f_1^{t-2} & \cdots & f_1^{t-h} \\ f_2^t & f_2^{t-1} & f_2^{t-2} & \cdots & f_2^{t-h} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_n^t & f_n^{t-1} & f_n^{t-2} & \cdots & f_n^{t-h} \end{pmatrix},$$

where the first column comprises all the objects at the current time t; the second column comprises all the objects at $t-1$, and so on and so forth throughout the matrix until the last column, which comprises the objects at $t-h$. Each object or FAD f_i^t is composed of two coordinates (latitude and longitude), which allows to represent the position of the FAD in the map: $f_i^t = (latitude_i^t, longitude_i^t)$. The variable h can be different for each FAD, and it depends on when it was released into the sea. In this case, having the last three positions of each FAD is enough (i.e., t, $t-1$ and $t-2$), so the FADs input matrix is as follows:

$$\begin{pmatrix} f_1^t & f_1^{t-1} & f_1^{t-2} \\ f_2^t & f_2^{t-1} & f_2^{t-2} \\ \vdots & \vdots & \vdots \\ f_n^t & f_n^{t-1} & f_n^{t-2} \end{pmatrix}$$

3.4.2.2 Vessels and fishing-information input

Given m vessels, the following inputs are needed:

- Initial position: $v_i = (v_{lat}, v_{lon})$, where $v_1 \neq v_2 \neq \ldots \neq v_m$.
- Average speed (s_i), in knots, when traveling from one object to the next. For simplicity, we use the same s_i for all vessels, but this value could be different in each case.
- Fishing time (FT) by object, with two possibilities:
  - The fishing time is the same for all objects: $FT_1 = FT_2 = \ldots = FT_n$.
  - The fishing time can be different for each object: $FT_1 \neq FT_2 \neq \ldots \neq FT_n$.

The fishing time could also be vessel-dependent instead of FAD dependent, using the same fishing time for all vessels ($FT_1 = FT_2 = \ldots = FT_m$) or a different fishing time for each vessel ($FT_1 \neq FT_2 \neq \ldots \neq FT_m$). For the sake of simplicity, our solution uses the same fishing time for all FADs.

3.5 Methodology

Based on historical data provided by GPS buoys tied to FADs, the paper proposes a new approach that combines a metaheuristic method with a predictive technique. We first estimate the trajectories of FADs and then run a genetic algorithm to determine the best possible route considering the FADs future locations. These steps are explained separately below in different subsections. The last one describes the final solution, which implements the GA with a multiple-trajectory prediction.

3.5.1 The predictive technique: estimating FADs future position

Once the information inputs are available, the next step is to predict the next position of each FAD. This challenge could be met with several methods that simulate and predict current movements in the sea (Özgökmen et al, 2000); however, most of them use complex mathematical models based on data that are difficult to obtain for a tuna vessel. On the one hand, FADs are hand-made objects, often built by fishermen themselves with a bamboo framework (about 3×1.5 m). Their manufacture therefore introduces a significant variability that makes the mathematical modeling of their movement very complex. On the other hand, underwater nets are commonly attached underneath these FADs. The length of these nets has progressively increased and can reach today a depth of 50 m in the Eastern Pacific (Fonteneau and Pianet, 2000). In any case, no two FADs can be assumed to drift identically under the same environmental conditions, which makes the modeling to predict future positions based on standard Lagrangian buoys a futile exercise.

Under these circumstances, a feasible alternative that needs little information and computational power is Newton's motion equation. It is thus used here to predict the future position of each FAD, because the last three positions of each object would be enough to obtain a rather accurate prediction in the short-term. In fact, the equation effectiveness decreases with time because the error has a cumulative effect. As soon as the buoys transmit their subsequent positions, however, the algorithm can update the last position and can be executed again to predict the best route. So if the prediction for a specific FAD is not accurate enough to predict the best set of routes, the solution can be updated when the next message is received, therefore showing a better optimal route if one exists. It is in this sense worth noting that the goal here is to predict where FADs will be in the near future (not only its next position, but many future positions). Newton's motion equation is thus applied between an initial point and a current or final point:

$$y_{t+1} = y_t + v_t \Delta_t + \tfrac{1}{2} a_t \Delta_t^2, \tag{3.1}$$

where

- y_t is the position at the end of the interval (displacement).
- v_t is the velocity at the end of the interval t.
- Δ_t is the time interval between the initial and current states.
- a_t is the acceleration at time t.

The variable t is considered discrete. Knowing the position of a FAD requires that the attached buoy transmits its coordinates through satellite communication. Since this is costly, the number of transmissions of each buoy is limited normally to twice a day. With this information, in order to calculate the future positions of an specific FAD (f_i) at time $t+x : x \in \{1,\dots,r\}$ we only need its last three positions at time $t, t-1$, and $t-2$.

$$\begin{aligned}
\hat{f}_{t+1} &= f_t + v_t \Delta_t + \frac{1}{2} a_t \Delta_t^2 \\
\hat{f}_{t+2} &= \hat{f}_{t+1} + \hat{v}_{t+1} \Delta_t + \frac{1}{2} \hat{a}_{t+1} \Delta_t^2 \\
&\vdots \\
\hat{f}_{t+r} &= \hat{f}_{t+r-1} + \hat{v}_{t+r-1} \Delta_t + \frac{1}{2} \hat{a}_{t+r-1} \Delta_t^2
\end{aligned}$$

It is important to note that previous predictions are used to calculate new prediction values, so it is easy to understand that results worsen as this simple approach makes more predictions. Long term predictions would accordingly need a different prediction method indeed.

3.5.2 Algorithm design

mTSP is a class of NP-hard Combinatorial Optimization (CO) problem. Complete algorithms are guaranteed to find every finite size instance of a CO problem, but might need exponential computation time in the worst-case scenario. Approximate methods, by constrast, do not ensure optimal solutions but alternatively offer good solutions in a significantly reduced amount of time (Blum and Roli, 2003b).

In the last 20 years, a new kind of approximate algorithms has emerged that basically tries to combine basic heuristic methods in higher level frameworks aimed at efficiently and effectively exploring a search space. These methods are nowadays commonly called metaheuristics, and refer to an iterative generation process that guides a subordinate heuristic by combining different concepts for exploring and exploiting the search space. Furthermore, learning strategies are used to structure information in order to find efficiently near-optimal solutions (Osman and Laporte, 1996).

There are many heuristic methods to solve optimization problems like mTSP. Some examples include Ant Colony Optimization algorithms (Mavrovouniotis and Yang, 2013; Kuo and Zulvia, 2017), Particle Swarm Optimization algorithms (Du and Li, 2008; Lynn and Suganthan, 2017), Simulated Annealing (Song et al, 2003) or Artificial Neural Networks (Yegnanarayana, 2009). Evolutionary algorithms (EA) are also a particular type of metaheuristic methods that are inspired by natural, self-organized systems and biological evolution. Examples of these algorithms include Genetic Algorithms (Deng et al, 2015; Li et al, 2017) or Artificial Immune Systems (Alonso et al, 2015; Banerjee, 2017), which -as any EA- are flexible in the sense that they can be adapted to changing environments by exploiting information from earlier moves. EAs are therefore especially suitable for optimization problems in dynamic environments such as the ones studied here (Wang et al, 2007; Zhou et al, 2003). Furthermore, the selected algorithm must be evolutionary for the following reasons (Zhou et al, 2003):

- EAs employ population policy and each individual in the population is an alternative solution for a dynamic TSP.
- The population policy allows individuals to hold diverse information. Population diversity has been proved to be a very important factor for species' existence in changing environments. By easily integrating some diversity-preserving techniques, individuals are also enabled in the algorithm to quickly fit the dynamic environments.

Within this context, GAs can be viewed as an evolutionary process whereby a population of solutions evolves over a sequence of generations to achieve a near-optimal solution. They can therefore be defined as computer programs that evolve in ways that resemble natural selection, solving complex problems that even their creators do not necessarily have to understand completely (Holland, 1992). They were first introduced by (Holland, 1975) to address optimization problems using techniques inspired by natural evolution (Winter et al, 1996), and this idea has led to many theoretical developments over the last 40 years (Reinelt, 1994; Smith and Smith, 2002). In fact, GAs represent one of the most consolidated approaches to TSP (Razali et al, 2011; Potvin, 1996), based on the following components (Srinivas and Patnaik, 1994):

- A genetic representation for the feasible solutions to the optimization problem
- A population of encoded solutions
- A fitness function that evaluates the optimality of each solution
- Generic operators that produce a new population from the existing population
- Control parameters

A population of solutions is maintained and a reproductive process allows parent solutions to be selected from the population. A *crossover* operator recombines portions from parent solutions to produce offspring, whereas a *mutation* operator maintains genetic diversity among solutions, preventing the GA to converge to a local minimum.

After using *crossover* and *mutation* on the initial population, the resulting offspring solutions exhibit some of the characteristics of each parent. The population components are then evaluated based on a given fitness function (in this case, the total distance traveled). Analogously to biological processes, the offspring with relatively good fitness levels are more likely to survive and reproduce, with the expectation that fitness levels throughout the population will improve as it evolves. Each solution is therefore composed of an array of numbers in which each number represents one of the targets in the route (the genes).

The potential of genetic algorithms (GAs) can be easily perceived in the wide use among scholars in such fields as Control engineering (Thomas and Poongodi, 2009), Economics (Chiang, 2005), Medicine (Kosakovsky Pond et al, 2006), Mechanical engineering (Bernardino et al, 2007), etc. In the case of route optimization, GAs have several applications such as ship routing with time deadlines (Karlaftis et al, 2009), vehicle routing with time windows (Wang and Regan, 2002; Baker and Ayechew, 2003; Garcia-Najera and Bullinaria, 2011), and VRP with loading constraints (Ruan et al, 2013). Although GAs are thus present in many route optimization problems, however, the TSP with dynamic targets is still an underdeveloped issue where GAs have hardly arrived.

GAs can nevertheless make an important contribution also in dynamic scenarios. To be sure, their near-optimal solution to a typical TSP problem is necessarily different from the one required when targets move. As explained above, however, GAs show an evolutionary behavior that allows to converge relatively fast towards a robust solution (Jih and Hsu, 2004; Bjarnadóttir, 2004). Coherently, the GA used here can be fed with the trajectory prediction of FADs in order to identify the best route that each vessel within the fleet must follow.

In this paper, the specific properties of the GA reflected the need to minimize the distance that the vessels traveled throughout the recovering process. We summarize these properties below and, in the following subsections, provide a brief report on the algorithm design:

- Population size: $1,200$.
- Natural-selection mechanism: tournament selection.
- Tournament size: $1,200$ individuals selected in couples.
- Crossover type: two-part chromosome approach with $p = 0.8$.
- Mutation type: simple *mutation* between two elements with $p = 0.2$.
- Fitness: total distance traveled by all the vessels.
- Stopping criteria: $3,000$ iterations with no any fitness improvement.

3.5.2.1 Solution encoding

Each solution of the mTSP is a vector divided into two parts:

$$(f_1, f_2, \ldots, f_n \mid z_1, z_2, \ldots, z_m) \tag{3.2}$$

The first part of the vector $(f_1, f_2, \ldots, f_n)$ represents all the targets (FADs) that must be visited, and the second part $(z_1, z_2, \ldots, z_m)$ shows how many targets each vessel must visit. Thus, n is the quantity of FADs to recover, and m is the number of vessels that need to pick up those n objects. Note that $z_1 + z_2 + \ldots + z_m = n$. Each FAD, therefore, will be a point on the route and will be represented by a number. For instance, $(9,4,7,3,5,1,2,6,8 \mid 3,4,2)$ is a possible solution for a scenario with nine FADs and three vessels. This solution vector means that the first vessel has to recover three objects, $[9,4,7]$; the second vessel has to gather four objects, $[3,5,1,2]$; and the third vessel has to collect two objects, $[6,8]$.

3.5.2.2 Initialization

The initial population is chosen randomly with the aim of covering the entire search space. A random set of $1,200$ initial solutions has been set, which covers the problem perfectly (Yang, 1997). The fitness of each solution is measured as the total distance that the vessels travel to recover all the FADs.

The same method used by (Carter and Ragsdale, 2006) is employed here to generate the initial population. The first part of each chromosome is a randomly generated permutation of n cities. The greedy solutions are subsequently generated by examining the present location of each (vessel) and then calculating the unassigned city (FAD) that is closest to each salesman. Once a city is assigned to the closest salesman, the process continues until all cities are assigned to a salesman. This gives the GA a good starting point in the search space and improves the final results.

Additionally, as the sum of the positive integers in the second part of the chromosome must equal n, the m gene values (x_i) for the second part of the chromosome is generated as a discrete uniform distribution of random numbers between 1 and $n - \sum_{k=1}^{i-1} x_k$, for $i = 1$ to m (i.e., the maximum value for each successive gene value is based on n and the sum of the previous values). These values are then randomly assigned to the genes in the second part of the chromosome.

3.5.2.3 Prediction

Using the inputs of n drifting objects with known values for the current location and previous two positions, the r future positions for each object are estimated through Newton's motion equation. At first, this prediction is performed only once. The value of r will depend of the number of targets n and vessels m. As n increases, the vessels will need more time to recover the objects, so r typically grows as n grows, and decreases when m grows. As the vessels' speed (s_i) grows, r becomes smaller. There must be a balance between n, m, s_i, and the number of future predictions (r) that we can calculate. If there were more predicted positions than the problem requires, the algorithm will only use the future positions that it really needs; however, r must be large enough to determine the solution.

$$\begin{pmatrix} f_1^t & f_1^{t-1} & f_1^{t-2} \\ f_2^t & f_2^{t-1} & f_2^{t-2} \\ \vdots & \vdots & \vdots \\ f_n^t & f_n^{t-1} & f_n^{t-2} \end{pmatrix} \rightarrow \begin{pmatrix} \hat{f}_1^{t+1} & \hat{f}_1^{t+2} & \cdots & \hat{f}_1^{t+r} \\ \hat{f}_2^{t+1} & \hat{f}_2^{t+2} & \cdots & \hat{f}_2^{t+r} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{f}_n^{t+1} & \hat{f}_n^{t+2} & \cdots & \hat{f}_n^{t+r} \end{pmatrix} \tag{3.3}$$

3.5.2.4 Fitness

The fitness measure is the total number of miles traveled by all the vessels; i.e., the total travel distance.

3.5.2.5 Selection

The tournament *selection* method of individuals is based on their performance. With an initial random set of 600 pairs, the couple-tournament is used for the mating selection, so each pair competes with the others. The 600 winners of each tournament are selected to be the parent in the next generation (or offspring). This selection pressure allows the GA to improve the population fitness through successive generations. The algorithm needs probability values for the *crossover* and *mutation* in order to do so. Once the *crossover* and selection are made the offspring size will be again $1,200$ individuals.

3.5.2.6 Crossover

In the *crossover* method, each offspring inherits characteristics from its parents. *Crossover* causes a structured, yet randomized exchange of genetic material between solutions, with the possibility that 'good' solutions can generate 'better' ones (Srinivas and Patnaik, 1994). For the mTSP, the *crossover* approach is different from the TSP due to the complexity of the problem, i.e., different vehicles recovering different objects. A two-part chromosome is commonly used, therefore, instead of a one-part chromosome.

The two-part chromosome technique, as the name implies, divides the chromosome into two parts. The first part with length n represents a permutation of n cities, whereas the second part with length m reflects the number of cities assigned to each salesman. The *crossover* operation for the two-part chromosome is separated into two independent operations. The first operation uses an ordered *crossover* operator, while the second operation uses an asexual *crossover* operator to ensure that the second part of the chromosome remains feasible.

The solution implemented here uses a two-part chromosome approach that improves the GA's search performance when solving the multiple TSP (Shuai et al, 2013). This method minimizes the size of the search space by reducing the redundant candidate solutions. The parameters of the GA were tested initially with the same probability (0.5-0.5); then the values were changed up and down by 0.1 until an optimum was reached in 0.8 and 0.2, respectively, for *crossover* and *mutation*. Lower *crossover* probabilities and higher values of *mutation* reduced the exploration capability of the GA and increased the amount of disruption to each gene in the solution. Similarly, higher *crossover* probabilities and lower *mutation* probabilities eliminate good candidates because of too much exploration and too much gene disruption, thus yielding worse results.

3.5.2.7 Mutation

Mutation is a genetic operator used to maintain genetic diversity from one generation of genetic algorithm chromosomes to the next one. These random changes will gradually add some new characteristics to the population that could not be supplied by the crossover, preventing the algorithm to avoid a local minimum when the population is too similar. The *mutation* probability, as noted above, has been set at 0.2.

There are many *mutation* types; in the simplest form, only one chromosome is mutated (bit string *mutation*), but there are more complex approximations as well (Flip bit, Boundary, Gaussian, etc.). This paper uses the simplest type, swap *mutation*, to randomly exchange two elements of the route. Notice, however, that this *mutation* is only applied for the first part of the chromosome; the second part continues without changing.

3.5.2.8 Execution of the algorithm and stopping criterion

The execution of the proposed algorithm must be carried out in a centralized way because the final, unique solution must be transmitted to all m vessels at the same time. The vessels' operators can all therefore know exactly which FADs have been assigned to each vessel. The algorithm cannot be executed on board because this is a metaheuristic approach, so the final solution could yield a different solution for each vessel. For this reason, the algorithm should be executed only once, either in the main office or on just one of the vessels. Results would be subsequently communicated to all vessels.

The execution of the algorithm is halted when, after $3,000$ loops, there is no fitness improvement. This number of maximum loops has been set so that the execution time of the GAMTP solution can vary from 1 to 10 min, depending on the number of FADs and vessels (n and m). This time constraint actually represents a fairly quick response since the company will organize the vessels' work for a week or more each time.

The algorithm has been developed in C# Programming Language and integrated into the MSB software, which is used by tuna vessels to receive the information from their buoys. It has been developed in a decoupled way where both, the collecting method (NN, GA or GAMTP) and the GAMTP parameters, can be easily changed. The solution might be useful for any type of tuna fishing company, regardless of the number of vessels or FADs. Furthermore, the algorithm can be specialized for specific tasks such as prioritizing the recovery of particular targets, working with time windows or finding the optimal speed of each vessel, which saves even more fuel. Concerning the computational experiments, they were carried out on a Windows 10 operating system with an Intel Core i7 running at 2.7 GHz and 16 GB of RAM.

3.6 Implementing the GA with a multiple-trajectory prediction

Based on the inputs and techniques described, this subsection explains how an evolutionary algorithm can be combined with a prediction technique (in this case, Newton's motion equation) to improve results when targets are constantly moving.

The addition of prediction is a bold departure from the initial state of the genetic algorithm, which evolves through multiple intermediate solutions to reach a final route for each of the vessels. Accordingly, the method described here, GAMTP (Genetic Algorithm based in Multiple-Trajectory Prediction), evolves from scratch to provide a set of routes based on the FADs predicted movement, which feeds the GA through the fitness-calculation process.

Algorithm 2 shows the pseudo-code of our GAMTP solution, and Algorithm 3 details how the fitness is calculated for each solution.

In Algorithm 2, different FADs and vessels positions are used as inputs to calculate the quasi-optimal best route. The first step is to predict r future positions for each FAD movement, so that the first set of solutions is created. Hence, following the rules of GAs based on the fitness of each solution, the population evolves towards the final route. It is worth noting that the singularity of the proposal with regard to existing literature is in how the fitness is calculated, as explained in detail in Algorithm 3.

The fitness algorithm calculates how well each solution matches the predicted FADs locations, as calculated previously ($\hat{f}^{t+r}$); this prediction is combined with the vessels' speed and fishing times to approximate the real route that each vessel is going to travel in the dynamic FAD context.

In order to determine the fitness of a given solution it is necessary to calculate the distance traveled by each vessel. The variables z_0, z_i and z_f are used to select the FADs that each vessel has been assigned to recover. Based on their speed, each vessel will recover its FADs at their estimated position, depending on their arrival time, which starts at t and evolves with $t+r$. Finally, the fitness of the solution is measured as the sum of the distance that all m vessels traveled ($\sum_{i=1}^{m} d_i$).

Note that we ignore a FAD movement whenever a vessel is traveling to recover it. The rest of the FADs, however, are assumed to continue moving while the vessels are traveling or fishing, as shown in the fitness calculation. The rationale of this mathematical simplification is to avoid calculating the collision vector from the vessel to the object when it is moving. The only challenge of the alternative approach has to do with the time calculation, which should improve the result of the final solution because it would be more accurate than the current time calculation.

Figure 3.2 shows how GAMTP solution works. It includes the vessels and FADs initial positions at time t, as well as the final positions of each FAD. Figure 3.2 is an example of how the final fitness would be calculated, as the positions where the vessels recover the FADs depend on the time of recovery. By selecting one vessel as starting

Data: n FADs positions (f_i) at time, $t, t-1$, and $t-2$:

$$\begin{pmatrix} f_1^t & f_1^{t-1} & f_1^{t-2} \\ f_2^t & f_2^{t-1} & f_2^{t-2} \\ \vdots & \vdots & \vdots \\ f_n^t & f_n^{t-1} & f_n^{t-2} \end{pmatrix}$$

Data: m vessels positions: $(v_1, v_2, \ldots, v_m)$
Data: Vessels speed (s) and fishing time for each FAD
Result: Final route for each vessel
for $i \leftarrow 1$ ***to*** n **do**
  using input $(f_i^t, f_i^{t-1}, f_i^{t-2}) \rightarrow$ calculation of the r future positions of each target f_i: $(\hat{f}_i^{t+1}, \hat{f}_i^{t+2}, \ldots, \hat{f}_i^{t+r})$;
end
GA initialization: k first solutions calculated;
for $i \leftarrow 1$ ***to*** k **do**
  fitness(i)
end
while *Stopping criteria not reached* **do**
  parent selection;
  new offspring generation: k solutions calculated;
  for $i \leftarrow 1$ ***to*** k **do**
    fitness(i)
  end
end
final route = best fitness;

Algorithm 2: GAMTP algorithm.

point, we can see that the first vector is to travel directly to the first FAD; however, the second iteration finishes at the FAD's expected location at time $t+x : x \in \{1, \ldots, r\}$. The same procedure holds for the rest of the objects.

Note that, when $f^t = f^{t-1} = f^{t-2}$, then $f^t = \hat{f}^{t+1} = \ldots = \hat{f}^{t+r}$. This means that, if there is no FADs movement, and if the fishing time is equal to zero, our GAMTP solution is identical to the classical GA approach for the mTSP. Thus, our contribution is a generalization of all the solutions, from the particular case when the targets are not moving and the vessels do not spend any time recovering targets, to the case where targets are moving and different fishing times exists. Thus, for the implementation of the classic mTSP based on GA, no target movement is assumed for the fitness calculation, so the resulting solution is compared to our GAMTP solution. This creates a static *vs.* dynamic comparison, through which it is easy to see that the total distance traveled using GAMTP and standard mTSP without prediction is very different.

Data: Solution input: $(f_1, f_2, \ldots, f_n \mid z_1, z_2, \ldots, z_m)$
Data: FAD position matrix:

$$\begin{pmatrix} f_1^t & \hat{f}_1^{t+1} & \hat{f}_1^{t+2} & \cdots & \hat{f}_1^{t+r} \\ f_2^t & \hat{f}_2^{t+1} & \hat{f}_2^{t+2} & \cdots & \hat{f}_2^{t+r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{f}_n^t & \hat{f}_n^{t+1} & \hat{f}_n^{t+2} & \cdots & \hat{f}_n^{t+r} \end{pmatrix}$$

Data: Vessels position inputs: $(v_1, v_2, \ldots, v_m)$
Data: Vessels speed (s) and fishing time (the same for all vessels)
Result: fitness of the solution $\equiv$ total distance traveled
variables initialization: $z_0 = 1$;
for $i \leftarrow 1$ ***to*** m **do**
  $v = v_i$: initial position of the vessel i;
  $r = 0, d_i = 0$: time and distance equal to zero;
  $z_f = (z_0 + z_i) - 1$;
  for $j \leftarrow z_0$ ***to*** z_f **do**
    nt = time to recover $\hat{f}_j^{t+r}$ at s knots from position v;
    nd = distance traveled from position v to $\hat{f}_j^{t+r}$ at s knots;
    v = predicted position of $\hat{f}_j^{t+r}$: update the vessel's position;
    $r = r + nt +$ fishing time: update the time;
    $d_i = d_i + nd$: update the distance;
  end
  $z_0 = z_i + 1$;
end
fitness $= \sum_{i=1}^{m} d_i$

Algorithm 3: Fitness algorithm.

3.7 Computational results

In this section we discuss the improvement achieved by addressing dynamic mTSP with the method proposed: GAs based on Multiple Trajectory Prediction (GAMTP). Initially we compare the Nearest Neighbor (NN) strategy, which is the method normally used by tuna vessels, with GAMTP. Then, we compare the performance of GAMTP method with mTSP solved only by genetic algorithms (GA), i.e., without prediction.

It is worth recalling that we use real data from different tuna fishing companies. In order to test our model, and with scientific purposes exclusively, Marine Instruments provided us with anonymous real data from several tuna vessels fishing in the FAO capture zone no. 57 (Eastern Indian Ocean) from April 9th to April 23rd. According to internal company records, in this area there are about 40 vessels operating at the same time. Taking into account a maximum retrieving of 14 FADs per vessel and week, it would be possible to collect a maximum of 1,120 FADs in this period of time. The sample used in our simulations consists of 150 randomly selected FADs using the MSB software (platform to receive and visualize the buoys data) from Marine Instruments. This quantity represents a percentage of 8.0% with a level of significance greater than 10% (sampling error of 7.45%). Regarding the working conditions of the simulations,

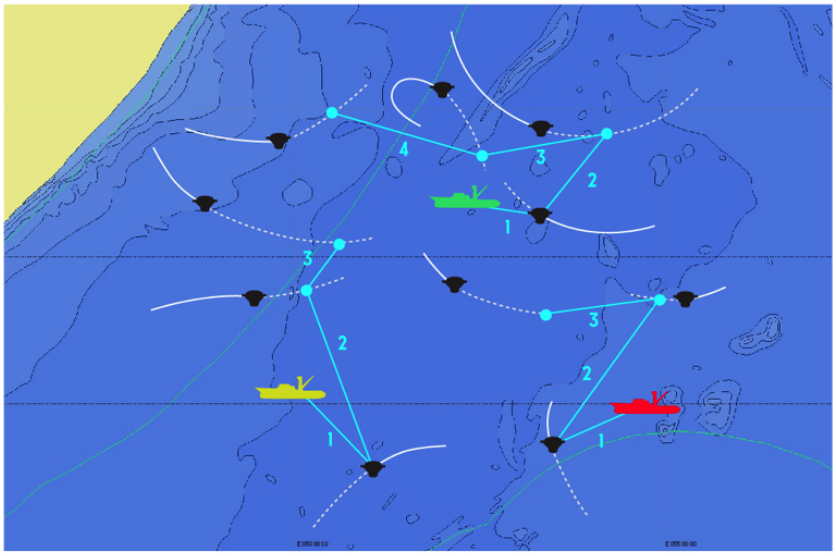

Figure 3.2: GAMTP algorithm procedure.

they are the following:

- Vessels speed (average): 12 knots.
- FADs speed = FADs have different speeds, randing from 0.2 knots up to 2 knots.
- Fishing time: 3 h for each object.
- Number of vessels: 2, 3, and 4 (the typical range per group in practice).
- Number of FADs: from 20 to 36.
- Number of FADs/vessel: from 5 (20 FADs for 4 vessels) to 14 (28 FADs for 2 vessels).

Considering this scenario, the total distance traveled by the group of vessels is calculated based on both the number of vessels (two, three or four) and the number of FADs (from 20 to 36). We have performed 10 measurements in each experiment, varying the positions of the FADs and the vessels, to obtain representative mean values for each case (Table A1 in Appendix describe the design of the simulation tests). Once the values for each of the experiments are obtained, the mean is calculated and results from GAMTP are compared with the NN strategy and GA (without prediction) respectively. The results (average, standard deviation and the improvement percentage achieved between each two methods) are shown in Table 3.2. We can observe that our GATP method is always better than the NN and GA for recovering from 20 to 36 FADs.

These results are supported statistically (Table 3.3 and 3.4). The comparison has been performed using a repeated-measures analysis of variance (repeated-measures ANOVA) for each experiment. Repeated-measures ANOVA must be used when, as here, we analyze differences in mean scores under three or more different conditions. One of the main conditions with these study designs is that the same individuals (vessels here) are being measured more than once on the same dependent variable (i.e. why

Table 3.2: Computational results.

			Total distance traveled (nautical miles)			Improvement Comparison	
Num. Vessels	Num. FADs		NN	GA	GAMTP	GAMTP vs NN	GAMTP vs GA
2	20	$\bar{x}$	12,084.7	10,764.4	10,211.6	15.5%	7.3%
		σ	2,118.7	1,601.2	1,384.6		
	24	$\bar{x}$	11,549.6	10,649.9	9,765.3	15.4%	8.8%
		σ	992.5	648.8	875.9		
	28	$\bar{x}$	13,600.9	12,597.8	11,778.4	13.4%	7.0%
		σ	975.8	901.6	567.5		
	32	$\bar{x}$	11,040.1	10,588.4	9,997.7	9.4%	5.6%
		σ	1100.5	891.2	603.7		
	36	$\bar{x}$	15,121.4	15,172.3	13,942.8	7.8%	8.1%
		σ	1713.0	1050.9	590.1		
3	20	$\bar{x}$	13,876.2	11,310.6	10,918.4	21.4%	5.1%
		σ	2,011.0	535.8	507.4		
	24	$\bar{x}$	14,091.9	12,193.4	11,615.3	17.6%	6.2%
		σ	1,424.1	518.8	426.0		
	28	$\bar{x}$	13,505.8	11,858.3	11,357.3	15.9%	5.0%
		σ	3,024.6	2,497.3	2,403.0		
	32	$\bar{x}$	12,273.5	11,367.6	10,446.8	14.9%	8.1%
		σ	1,178.3	605.1	884.2		
	36	$\bar{x}$	11,142.6	10,418.8	9,628.5	13.6%	7.6%
		σ	989.2	667.8	761.0		
4	20	$\bar{x}$	13,197.0	10,279.2	9,662.6	26.8%	6.7%
		σ	1,364.0	425.6	488.4		
	24	$\bar{x}$	13,666.9	11,202.4	10,506.6	23.2%	7.0%
		σ	1,742.6	342.2	555.2		
	28	$\bar{x}$	15,526.4	13,151.2	12,381.4	20.2%	5.9%
		σ	994.9	464.7	490.6		
	32	$\bar{x}$	13,897.0	12,572.3	11,710.7	15.7%	6.9%
		σ	1,412.1	519.4	528.6		
	36	$\bar{x}$	13,607.7	12,260.1	11,552.2	15.1%	5.8%
		σ	980.4	618.3	477.4		

it is called repeated measures) and the independent variable has different categories (here, the number of vessels per group). According to this, Table 3.3 shows significant differences among the methods used ($\text{Sig.} = 0.000 < 0.05$), and the results are consistent since the four most widely used tests for ANOVA (Phillai's trace, Wilks' Lambda, etc.) all indicate a very high significance. Secondly, getting deeper into how the means of the three methods are significantly different, Table 3.4 supports our descriptive analysis based on a Pairwise comparison: the average GAMTP distances are statistically lower than those obtained with NN and GA for the three group of vessels.

Table 3.3: Multivariate test (ANOVA)

Experiment	Type	Effect	Value	F	Hipothesis df	Error df	Sig.
Exp. 1:2 vessels	Method	Pillai's Trace	0.884	98.597	2.00	26.00	.000
		Wilks' Lambda	0.116	98.597	2.00	26.00	.000
		Hotelling's Trace	7.584	98.597	2.00	26.00	.000
		Roy's Largest Root	7.584	98.597	2.00	26.00	.000
Exp. 2:3 vessels	Method	Pillai's Trace	0.845	70.924	2.00	26.00	.000
		Wilks' Lambda	0.155	70.924	2.00	26.00	.000
		Hotelling's Trace	5.456	70.924	2.00	26.00	.000
		Roy's Largest Root	5.456	70.924	2.00	26.00	.000
Exp. 3:4 vessels	Method	Pillai's Trace	0.866	83.682	2.00	26.00	.000
		Wilks' Lambda	0.134	83.682	2.00	26.00	.000
		Hotelling's Trace	6.437	83.682	2.00	26.00	.000
		Roy's Largest Root	6.437	83.682	2.00	26.00	.000

In brief, the GAMTP strategy yields better results for all experiments (two, three, and four vessels) relative to other common optimizing strategies (Figure 3.3). We prove

Table 3.4: Pairwise comparison (ANOVA)

Num. Experiment	Method (a)	Method (b)	Mean Diff. (a-b)	Std. error	Sig.	95% Confidence Interval Lower Bound	Upper Bound
Exp. 1:2 vessels	NN	GA	1,074.37	206.848	.000	546.424	1,602.363
		GAMTP	1,941.233	171.268	.000	1,504.068	2,378.378
	GA	NN	-1,074.37	206.848	.000	-1,602.363	-546.424
		GAMTP	866.867	106.990	.000	593.742	1,139.916
	GAMTP	NN	-1,941.233	171.268	.000	-2,378.378	-1,504.068
		GA	-866.867	106.990	.000	-1,139.916	-593.742
Exp. 2:3 vessels	NN	GA	2,037.196	267.707	.000	1,353.884	2,720.508
		GAMTP	2,650.947	281.720	.000	1,931.869	3,370.025
	GA	NN	-2,037.196	267.707	.000	-2,720.508	-1,353.884
		GAMTP	613.751	59.118	.000	462.854	764.648
	GAMTP	NN	-2,650.947	281.720	.000	-3,370.025	-1,931.869
		GA	-613.751	59.118	.000	-764.648	-462.854
Exp. 3:4 vessels	NN	GA	2,585.811	234.998	.000	1,985.989	3,185.634
		GAMTP	3,334.864	256.734	.000	2,679.565	3,990.172
	GA	NN	-2,585.811	234.998	.000	-3,185.634	-1,985.989
		GAMTP	749.057	102.720	.000	486.868	1,011.246
	GAMTP	NN	-3,334.868	256.734	.000	-3,990.172	-2,679.565
		GA	-749.057	102.720	.000	-1,011.246	-486.868

therefore that integrating forecasting within a metaheuristic method, such as genetic algorithms, can yield better results than their simple non-predictive version in a dynamic scenario. Undoubtedly, however, to achieve even better results, the prediction method should be adapted to the specific characteristics of each environment. Here, for example, improving the accuracy of long-term forecasting for FAD trajectories would require a prediction technique that contemplates the ocean currents, winds or the Coriolis Effect, among other factors. In fact, the results comparison suggests that the relative improvement decreases as the collection time increases (i.e., the ratio FADs/vessel).

Number of FADs collected by vessel (FADs/Vessel)

Vessels | FADs	20	24	28	32	36
2	10	12	14	16	18
3	7	8	9	11	12
4	5	6	7	8	9

First comparison: GAMTP vs NN (a)

Vessels | FADs	20	24	28	32	36
2	15.5%	15.4%	13.4%	9.4%	7.8%
3	21.4%	17.6%	15.9%	14.9%	13.6%
4	26.8%	23.2%	20.2%	15.7%	15.1%

Second comparison: GAMTP vs GA (b)

Vessels | FADs	20	24	28	32	36
2	7.3%	8.8%	7.0%	5.6%	8.1%
3	5.1%	6.2%	5.0%	8.1%	7.6%
4	6.7%	7.0%	5.9%	6.9%	5.8%

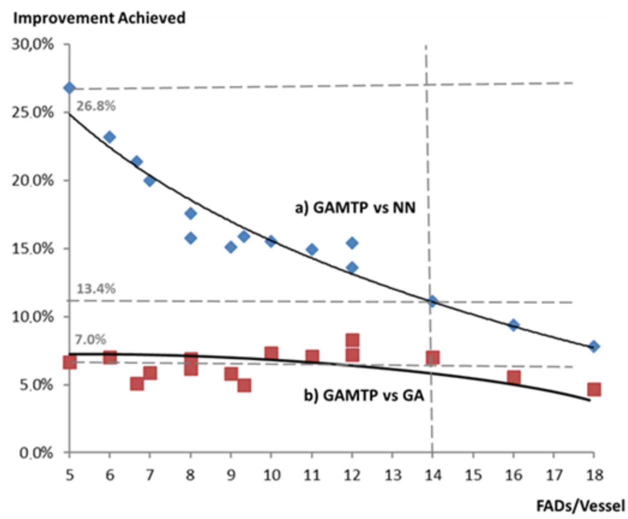

Figure 3.3: GAMTP improvements evolution as the ratio of FADs/vessel increases (built from Table 2).

Figure 3.3 describes this situation. First, from Table 3.2, the FADs/vessel ratio and improvement reached for each experiment were calculated (tables on the left); then these values were ordered according to this ratio and plotted. Figure 3.3 shows how lower collection rates (5 FADs/vessel) result in GAMTP results being as much as 26% better than NN results, and simultaneously verifies that for higher working ratios (14 FADs/vessel), this improvement is reduced up to 13%. A similar behavior shows the comparison with GA. Even in those simulations with 16 or 18 FADs/vessel (32 and 36 FADs for 2 vessels) results indicate that the GAMTP could reach a worse performance than the procedure without prediction (Figure 3.3). While this industry never reaches ratios as high as 18 FADs per vessel in this period of time, these simulations allow us to enrich and understand better the limitations of this proposal. This is due to the weakness of Newton's motion equation when making predictions in the long term, which obviously aggravates results as the number of FADs that each vessel retrieves increases. This reinforces the importance of prediction in the algorithm: although GAMTP yields better results than common strategies, the improvements decrease as the number of FADs increases.

3.8 Conclusions

This paper proposes a new method to address the multiple traveling salesman problem with moving targets (mTSP-MT). Unlike previous analyses, where the focus was on ad hoc quasi-experiments, moving targets are addressed here using an unrestricted generic solution that combines a metaheuristic method with a predictive technique. Particularly, the method first estimates the trajectory of each target using Newton's motion equation to feed then a GA that searches for the optimal route based on the total distance traveled by all salesmen. Hence the name "Genetic Algorithm based in Multiple Trajectory Prediction" (GAMTP). Based on historical GPS data for tuna fishing FADs, results show that the total distance traveled is always shorter with GAMTP than with other common methods used by ship-owning companies, such as NN or GAs without prediction. Integrating forecasting within a metaheuristic method (e.g., genetic algorithms) therefore yields better results than in the simple non-predictive version.

From an academic perspective, GAMTP can be seen as a generalization of classic approaches for solving the mTSP problem. With static objects, the predicted next position is the same in GAMTP and GA. However, in dynamic scenarios with moving objectives, GAMTP diverges because it evolves and continues optimizing the route by considering the future movement of each target. GAMTP could thus be used generically (in static and dynamic situations), as a suitable solution to obtain near-optimal routes.

In general terms, therefore, the paper opens up a set of possibilities for a wide range of real-world situations. GAMTP allows not just tuna vessels but any group of agents following moving targets (e.g., UAVs, autonomous devices and surveillance vehicles) to minimize the total distance traveled. Furthermore, within each of these possible fields of utilization, the algorithm could be specialized for specific tasks, such as prioritizing the recovery of particular targets (i.e., the ones that have more tuna beneath them), working with certain time windows for FAD recovery (e.g., tuna vessels do not fish at night), or finding the optimal speed of each vessel, which saves even more fuel. This should also open a new space for research. Furthermore, other interesting adaptations could be found in static scenarios with fixed destinations but where forecasting is especially relevant (e.g. situations where wind, currents, traffic jams, etc., are a critical factor). Similarly it is worth noting that GAMTP would allow to address the new potential uses that are emerging in mobility (e.g., delivery services, real-time mobility requirements, drones scheduling and collaboration, etc.) conducted by companies such as Uber and Amazon.

From an economic point of view, GAMTP can produce a considerable reduction in fleets' direct costs. In a typical situation with a fleet of 3 vessels, each working 24 h/day for 5 days, consumes approximately 1,000 USD/h (Parker et al, 2015). According to this, the average savings per campaign would be up to 64,800 USD. In addition, the reduction of the distance traveled not only affects the consumption of fuel with the estimations just shown, but also facilitates a more efficient utilization rate of the vessels thanks to a faster recovery process. Traveling is waste from an operational excellence perspective, whereas fishing, the output to maximize, is directly related to the number of FADs recoveries. It is worth noting, however, that given the chaotic nature of ocean currents and despite of the sophistication of the forecasting method, the GAMTP solutions add less value for long-term predictions. In fact, as shown in Figure 3.3, lower collection rates (e.g. 5 FADs/vessel) have greater improvements over standard techniques than do higher ratios (18 FADs/vessel).

Finally, taking into account the urge to stimulate sustainable business operations, the use of the GAMTP strategy in fishing companies would directly reduce CO_2 emissions by and average of 18% at current rates (considering the distance saved by a tuna vessel). This is undoubtedly a very significant improvement given that climate change represents one of the main challenges for humanity today (Howard-Grenville et al, 2014).

Acknowledgements

The authors wish to thank Marine Instruments for the buoy data. We also thank Iago Paz and participants at ACEDE 2018 for several useful suggestions. This research benefited from the financial support of the Spanish and Galician Governments through -respectively- grants ECO2016-76625-R and GRC2014/021.

3.A Appendix

Table A1: Experiment design

Num. Vessels	**Num. FADs**	**Num. Tests**	**NN**	**mTSP-GA**	**GAMTP**
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
2	20	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
2	24	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
2	28	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
2	32	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
2	36	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
3	20	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
3	24	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
3	28	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
3	32	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
3	36	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
4	20	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
4	24	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
4	28	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
4	32	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10
		Test 1	Total distance traveled 1	Total distance traveled 1	Total distance traveled 1
4	36	...	...	...	...
		Test 10	Total distance traveled 10	Total distance traveled 10	Total distance traveled 10

4

Optimization of logistic routes through information sharing policies: A game theory-based approach

En este capítulo abordamos el estudio de la optimización logística desde una perspectiva diferente a lo que venimos desarrollando a lo largo de esta Tesis, pero que a su vez es más amplia.

Si en los capítulos anteriores hemos desarrollado el problema de la optimización logística desde un punto de vista técnico, en esta ocasión trasladaremos la problemática a un punto de vista más abstracto, relacionado con el comportamiento de los agentes, que son los que toman parte en las decisiones operativas.

El trabajo, centrado en el caso concreto de la industria atunera tropical que pesca con objetos o FADs, se centra en modelizar el comportamiento de los agentes que intervienen en la toma de decisiones a la hora de pescar, que en este caso son los patrones de pesca por un lado y la compañía por el otro.

El objetivo es entender por qué muchos patrones son reacios a compartir los objetos con otros barcos, lo cual permitiría una optimización logística en términos globales.

Para modelizar el comportamiento de cada uno de los agentes se usa el marco de la Teoría de Juegos. En base a esto se calcula la función de utilidad de cada uno de ellos y se estudia matemáticamente, usando el modelo de Aumann.

Los resultados teóricos muestran que en el escenario actual de trabajo no existen incentivos para que los patrones compartan objetos. Estos resultados son contrastados empíricamente, en base a datos reales provenientes de la flota.

Tras el resultado alcanzado, se plantea el diseño de un escenario con compensaciones. La idea es animar a que los patrones decidan compartir sus objetos. El nuevo escenario es validado teóricamente y contrastado a través de simulaciones con datos reales.

Los resultados muestran mejoras importantes para la flota, pues el simple hecho de que los barcos compartan la información de sus objetos permiten que se produzca una optimización de la ruta global de la flota, permitiendo ser más eficientes y reduciendo las emisiones de CO_2 a la atmósfera.

La forma novedosa en la que se aborda el problema de optimización logística usando la Teoría de Juegos, abre una puerta al uso de esta técnica para el diseño de políticas adecuadas en otros ámbitos, ya sea de la empresa u otros.

Este trabajo, en coautoría de Antonio Sartal y Gustavo Cid Bergantiños, ha sido enviado al journal *'Marine Policy'* en Septiembre de 2018, encontrándose en estos momentos a la espera de respuesta.

4.1 Abstract

Regulatory restrictions about the maximum number of FADs by vessel have been established in the tuna fishing industry during the last years, which incorporate new constraints on tuna fishing companies that need, however, to make profitable their high inversions. Based on real-data and argued with the scope of game theory, we address a new way of working for these companies related to the use of FADs between vessels, proving that sharing FADs maximizes both the fuel and time to entire fleets. Our findings show that, with the correct incentives, all stakeholders –company, skipper, and environment– can improve their results jointly when information is shared.

4.2 Introduction

The performance of the tropical tuna fishing industry is, more than ever, bound to the use of drifting fish aggregating devices (FADs), which have become widespread since 1991 (Ariz et al, 1992). With the scope of new regulations affecting the tuna industry, this paper addresses a study from the point of view of efficiency using game theory as theoretical framework.

The global tuna fishery is one of the largest in the world. The most widely used and fastest-growing fishing gear for targeting tuna is the purse seine (PS). Tropical PS started to operate in the Atlantic Ocean in the 1960s and were introduced into the Indian Ocean in the early 1980s. The tuna species are skipkack (*Katsuwonus pelamis*), yellowfin (*Thunnus albacares*), and bigeye (*Thunnus obesus*), and they tend to associate with objects floating at the surface of the ocean (Fonteneau and Pianet, 2000; Castro et al, 2001). The aggregate behavior of tuna with floating objects was first observed with natural floating objects (FOBs) from river mouths. With the aim of imitating FOBs, fishers started deploying large numbers of their own FOBs. These human-made drifting FADs generally consisted of bamboo with large pieces of net hanging below for stability in the surface currents, and they would stay adrift for up to 2 months (Ménard et al, 2000).

The increasing use of FADs concurrently resulted in apparent increases in PS catches per unit effort (CPUE) over time (Maufroy et al, 2016; Fonteneau et al, 2013). The extensive use of FADs by the PS fishery industry increases the possibility of a number of negative impacts, including a reduction in yield per recruitment of target tuna species, increased by-catch and perturbation of the pelagic ecosystem balance, and alteration of the normal movements of the species associated with FADs (Bromhead et al, 2003; Fonteneau and Pianet, 2000), however these effects are difficult to estimate (Lopez et al, 2014).

Due to the increased use of FADs, recent efforts from regional fisheries management organizations (RFMOs) have produced regulations on the number of FADs that a PS can manage. Other restrictions affect the global marine fisheries, like the minimization of bycatch and discards (Gilman, 2011; Zeller et al, 2018).

With these newly implemented restrictions, it is mandatory that the tuna fishing industry optimize the use of FADs. Although many studies have been published regarding the use of FADs and their implications, little research exists in how to help the tuna fishing industry optimize their fishing practices (Groba et al, 2015).

In this context, our work addresses a new way of working for the tuna fishing companies related to the use of FADs between vessels, proving that sharing FADs maximizes fuel efficiency and use of time and decreases CO_2 emissions across entire fleets. First, with a foundation in game theory and the well-being assessment in particular, two different theoretical mechanisms were developed: one without compensation and the other with compensation. The first mechanism (without compensation) explains why many vessels do not like sharing FADs, and the second mechanism (with compensation), shows an equilibrium where all vessels want to share FADs. Second, this theoretical approach is evaluated empirically with real-data through simulations, and the expected result emerges: There is a situation in which all the players –company and skippers– win, proving that best route optimization takes place when FADs information is shared between vessels. Data for this study come from different groups of tuna vessels retrieving their FADs in the Indian Ocean during April 2017.

The paper is organized as follows. The next section provides a review of the literature. Section 3 describes the game theory approach. Section 4 introduces the data, the experimental design and discusses the results. Finally, Section 5 concludes by highlighting the paper's main contributions and implications.

4.3 Background

The use of FADs by PS has had evolved over the years to improve fishing efficiency. The FAD itself has undergone improvements in shape, materials, and the lengths of nets, both to drift with the currents of interest and also for minimizing the risk of entangling turtles and other non-targeted species (Girard et al, 2004). FADs have also evolved technologically. Since the beginning, the use of artificial FADs has relied on tracking buoys to know where the FADs are. The first buoys were radio-based, and each vessel used secret frequencies to locate its own floating objects (Ménard et al, 2000). In 1996, GPS buoys with a virtually unlimited range appeared on the market and positively affected the expansion of fishing areas (Morón et al, 2001).

During the 2000s, satellite technologies, including Inmarsat D+ and Iridium SBD, became affordable alternative for the buoys (Moreno et al, 2007). The use of satellite communications was a revolution for the tuna fishing industry because, although each buoy had a monthly airtime fee, there were many advantages compared to the previous buoys based on radio technology. Some of these advantages included receiving FAD positions at any distance and commanding the buoys from the vessel. Further, satellite buoys did not use large carbon antennas for transmission like the radio buoys did. Additionally, satellite buoys were difficult to detect by radar, making them less likely to be stolen, which was a big advantage over radio buoys. Finally, satellite communication technology provided vessels the capacity to share their FADs positions with other vessels, which allowed vessels to work together with the aim of improving their fishing efficiency.

The last buoy improvement was the development of echo-sounders, which were introduced around 2008 to monitor the amount of biomass aggregated beneath the FAD (Lopez et al, 2014). This new technological device reduced the searching time (i.e., enables remote identification of FADs with associated tunas) and provided new information for fishers to learn more about the location and behavior of tuna and other associated species. Several indicators suggest that echo-sounder buoys could be as important or more important than other significant technological developments in the fishery industry, such as the introduction of sonar (Lopez et al, 2014).

The echo-sounder technology embedded in the buoy was a game-changer for the tropical tuna industry in terms of optimization. Before this improvement, PS traveled from FAD to FAD searching for tuna. After the introduction of the echo-sounder tuna buoys, they only traveled to FADs that had fish beneath, which improved their fishing efficiency by saving time and fuel and by discovering new fishing areas. Such efficiency made PS want more echo-sounder buoys in the water (then FADs), with the aim of constant fishing.

The consequences of this were an increase in the use of FADs. For example, in the Atlantic Ocean (Fonteneau et al, 2015), the total number of FADs increased 730%, from $1,175$ FADs active in January 2007 to $8,575$ in August 2013. In the Indian Ocean this number increased 458%, from $2,250$ FADs in October 2007 to $10,300$ FADs in September 2013 (Maufroy et al, 2016). This increase has resulted in regulation from RFMOs of the number of FADs that a PS can manage, for example, in the Indian Ocean, where the number of FADs as defined in Resolution 15/08, paragraph 7, will be no more than 350 active instrumented buoys and 700 acquired annually instrumented buoy per vessel per year (IOTC circular 2017-061). Currently, this limitation is also being followed in the Atlantic Ocean through the International Commission for the Conservation of Atlantic Tunas (ICCAT), and all indicators predict that the Pacific Ocean will follow the same initiative via the Inter-American-Tropical-Tuna-Commission (IATTC) (Fonteneau and Pianet, 2000).

Regarding PS activity with FADs, it is noteworthy that skippers have important economic incentives depending on how many tons they fish. Meanwhile, tuna fishing companies or firms pay these incentives with the aim of maximizing the number of fished tons of the whole company. The costs of the entire fleet are assumed by the firm, including salaries, goods, and fuel, among others.

In terms of fishing management and efficiency, Salas and Gaertner (2004) showed how essential it is for effective management to know the dynamics of the fisheries. Bez et al (2011) used a vessel monitoring system (VMS) to measure tuna fishing efforts to study and quantify the spatial dynamic of the tropical tuna PS fishing activity. In terms of fuel consumption, Parker et al (2015) analyzed fuel performance and the carbon footprint of the global PS tuna fleet. Meanwhile Hospido and Tyedmers (2005) employed life cycle assessment (LCA) to quantify the scale and importance of emissions that result from the range of industrial activities associated with contemporary Spanish PS fisheries. Gaertner and Dreyfus-Leon (2004) analyzed the shape of the relationship between CPUE and abundance in a tuna PS fishery, using a simulation with artificial neural networks. In terms of fuel consumption efficiency, Groba et al (2015) showed how important it can be to optimize the route of a tuna vessel retrieving FADs.

There are different studies in terms of efficiency in different fisheries, for example Belhabib et al (2018) compared the artisanal fisheries versus the industrial fisheries, showing that the catches per unit effort (CPUE) of the artisanal fisheries was 11 times lower than industrial CPUE. Guijarro et al (2017) studied the bottom trawl fishery in order to improve the efficiency in the Western Mediterranean. Another example is given by Rust et al (2017), discussing the excess capacity and efficiency in the quota managed Tasmanian Rock Lobster Fishery, or the efficiency in the sardinian fisheries cooperatives (Madau et al, 2018).

In the case of tropical tuna fishery efficiency, the literature is scarce. For this reason, and with recent RFMO regulations in mind, the proper use of FADs by tuna vessels is a matter of great importance for tuna fisheries.

In this paper, tuna fishing vessels behaviours using FADs is studied for first time from the point of view of game theory. Indeed, through real-data simulations, this paper shows that there are policies that change ways tuna vessels work with FADs. If tuna fishing companies settle on these new policies they could improve their overall efficiency.

4.4 The tuna fishing vessels problem: A game theory approach

4.4.1 The tuna fishing vessels problem

Tuna skippers have important economic incentives that directly depend on how many tons they fish. These incentives also depend on the tuna ton price (Jeon et al, 2008). Because of this, it is important for skippers to maximize the number of ton fished; the more a vessel fishes in less time, the better. A tuna vessel faces, for instance, an optimization problem regarding which route to follow to increase fishing using its FADs, which drift in the ocean (Groba et al, 2015).

Each vessel has its own limited number of FADs to fish. Tuna vessels usually work in small groups, of two or three fishing vessels assisted by a supply vessel to deploy and retrieve FADs from the ocean (Arrizabalaga et al, 2001). In these cases, FADs are shared among the group of fishing vessels, and incentives are shared as well. Groups of vessels that fish together are share confidence, which is one reason they are typically small.

By contrast, firms want to maximize the overall company profits, which means that vessels have to fish as much as they can and that variable costs, such as fuel, crew costs, and equipment must be minimized. This means optimizing fishing of n vessels (vessels that the firm owns) with m FADs (the sum of FADs from all the vessels of the company).

In this scenario, there is a trade-off between firm's and the skippers' interests, highlighted by the limitation of FADs as dictated by RFMOs. Facing this situation, a new scenario for analysis and improvement appears, hinging on how to maximize the profits for all agents. The aim of this paper is to study this equilibrium in detail with real-data and explain how and why tuna fisheries currently operate. Further, this paper presents a new proposal for FAD sharing policies, which shows improvements for both individual and collective performance and reduction of CO_2 emissions.

4.4.2 The tuna fishing vessels problem

Tuna skippers have important economic incentives that directly depend on how many tons they fish. These incentives also depend on the tuna ton price (Jeon et al, 2008). Because of this, it is important for skippers to maximize the number of ton fished; the more a vessel fishes in less time, the better. A tuna vessel faces, for instance, an optimization problem regarding which route to follow to increase fishing using its FADs, which drift in the ocean (Groba et al, 2015). Each vessel has its own limited number of FADs to fish. Tuna vessels usually work in small groups, of two or three fishing vessels assisted by a supply vessel to deploy and retrieve FADs from the ocean (Arrizabalaga et al, 2001). In these cases, FADs are shared among the group of fishing vessels, and incentives are shared as well. Groups of vessels that fish together are share confidence, which is one reason they are typically small. By contrast, firms want to maximize the overall company profits, which means that vessels have to fish as much as they can and that variable costs, such as fuel, crew costs, and equipment must be minimized. This means optimizing fishing of n vessels (vessels that the firm owns) with m FADs (the sum of FADs from all the vessels of the company).

In this scenario, there is a trade-off between firm's and the skippers' interests, highlighted by the limitation of FADs as dictated by RFMOs.

Facing this situation, a new scenario for analysis and improvement appears, hinging on how to maximize the profits for all agents.

The aim of this paper is to study this equilibrium in detail with real-data and explain how and why tuna fisheries currently operate. Further, this paper presents a new proposal for FAD sharing policies, which shows improvements for both individual and collective performance and reduction of $\mathbf{CO_2}$ emissions.

4.4.3 A game theory approach

A theoretical model was introduced to study the problem described previously. We considered two different mechanisms that the firm can use to incentivize vessels to share FADs. When vessels share FADs, the total distance traveled by all vessels is reduced, which produces cost savings for the firm. Our analysis was conducted through a non-cooperative game with incomplete information following the model of Aumann (1976), which we believe is the most suitable for this case. We also consider the Bayesian Nash equilibria (BNE) (Nash, 1951), the most standard solution for these kind of games (Harsanyi, 1967).

Let $N = \{1, \ldots, n\}$ the set of tuna vessels, briefly, vessels. We assume that all tuna vessels work for the same firm, which we denote by f.

There is a finite number of FADs (or buoys) that have been assigned to the vessels following some criteria. We assume that each FAD is assigned to a single vessel. Thus, each vessel $i \in N$ has an initial endowment $b_i = \left\{\left(b_i^k\right)\right\}_{k=0}^{n_i} = \left\{\left(x_i^k, y_i^k\right)\right\}_{k=0}^{n_i}$. The interpretation is the following. Vessel i has been assigned to handle n_i FADs $\left\{b_i^1, \ldots, b_i^{n_i}\right\}$. The position of each FAD k with $k = \{1, \ldots, n_i\}$ is given by $\left(x_i^k, y_i^k\right)$ where x_i^k denotes the latitude and y_i^k the longitude. Besides we denote by $b_i^0 = \left(x_i^0, y_i^0\right)$ the position of vessel i at the beginning of the process. We also assume that FADs are numbered in the order of recovering by vessel i. Namely, vessel i is located in position $\left(x_i^0, y_i^0\right)$. Thus, it moves to FAD b_i^1 and recover the tuna in such FAD. Next, vessel i moves to position FAD b_i^2 and so on.

Therefore, we make the following assumptions:

- Each vessel knows the position of all FADs to which has been assigned. Each vessel does not know the location of the FADs assigned to other vessels.

- In the theoretical model, we assume that each vessel has a cost c per mile traveled between FADs. This cost is paid by the firm. In our simulations, we compute c by assuming that the average vessel speed is 15 knots. Thus, we estimate a fuel cost of $29 US per nautical mile traveling between FADs.

- Vessels cannot know in advance the amount of tuna they will find at each FAD. We denote by q the expected amount of tuna by FAD. We denote by q_i^k the amount of tuna recovered by skipper i in FAD b_i^k. These amounts will be known only after fishing.

  In our simulations we take $q = 6.1$ tons for every skipper i and every FAD b_i^k.

- Each vessel i cannot know in advance the amount of time t_i^k for recovering the tuna of FAD b_i^k.

  In our simulations we assume that t_i^k is 3 hours for every vessel i and every FAD b_i^k.

- Each skipper receives a price p by each amount of tuna fished.

  In our simulations we consider several values for p.

Once vessel (skipper) i has recovered all of its FADs, the utility obtained is computed as the amount fished multiplied by the price paid by the firm. Namely,

$$p \sum_{k=1}^{n_i} q_i^k$$

The utility of the firm is

$$(p_f - p) \sum_{i=1}^{n} \sum_{k=1}^{n_i} q_i^k - c \sum_{i=1}^{n} d(b_i)$$

where $d(b_i)$ is the distance traveled by vessel i for recovering all FADs in b_i. Namely, the firm pays a unit price of p to every vessel and sells the fish at the price p_f. Additionally, the firm has to pay costs associated with the travel of the vessels.

As skippers do not pay for fuel, they do not have an incentive to share their FADs to minimize the distance traveled. Nevertheless, the firm has incentives. If the cost is reduced (and all FADs are recovered), then the total utility of the firm will be increased.

We consider two possible mechanisms that firms can use to induce skippers to share their FADs. We model such mechanisms as two games with incomplete information following Aumann's model. Additionally, we study the Bayesian Nash equilibria (BNE) of both games, which provide predictions of the behavior of rational agents when facing such situations.

In the Appendix, we theoretically study both mechanisms, and we formally present the games for modeling both mechanisms. We also compute the BNE associated with both mechanisms (Propositions 1 and 2).

For now, we present the results in a more informal way. The basic idea of both mechanisms is the same. First, the vessels or skippers decide independently if they want to share its FADs or not. If a vessel says no, then this vessel fishes with its FADs. For the vessels that say yes, the firm reassigns their FADs among the cooperating. Next, every vessel fishes in its reassigned FADs.

Mechanism 1: Reassigning FADs without compensation. The firm will pay to the skippers according to the FADs each vessel has been assigned. Suppose that vessel i initially had 20 FADs, decided to share its FADs, and was reassigned with 18 FADs. The firm will pay skipper i according to the amount of fish obtained by the 18 reassigned FADs. If vessel i is reassigned with the same or more FADs than it initially had, then vessel i will be paid also according with the number of assigned FADs.

In Proposition 1 of the Appendix we theoretically study this mechanism. Here we discuss the practical implications of Proposition 1. According to part (a), if each vessel decides not to share its FADs (as in Example 2 of the Appendix), then we have a BNE, and the firm cannot save in fuel. In other cases (as in Example 1 of the Appendix), there could exist a different BNE, wherein some vessels share FADs and the firm saves fuel costs.

By part (b) of Proposition 1, we realize that the utility of each skipper i in any BNE will always be the same and coincide with the utility skipper i obtains when it does not share FADs. This result is independent of the number of FADs, the position of the FADs, and the information the vessels have over the position of the FADs. Thus, skipper i does not have an incentive to share its FADs in any circumstance because skipper i cannot improve its expected utility by sharing instead of not sharing. If skipper i shares its FADs, it could be the case that skipper i receives more FADs than it initially had, but it could also receive less. The average will be the same.

Our theoretical results prove that under this mechanism, skippers do not have incentives to share their FADs under any circumstance. This helps explain why tuna vessels work alone or in small groups. Nevertheless, this mechanism is not the most beneficial for the firm.

Mechanism 2: Reassigning FADs with compensation. The firm will guarantee to skippers that share their FADs to pay, at minimum, according with the number of FADs the vessel initially had. For example, suppose that vessel i initially had 20 FADs, decided to share its FADs, and is reassigned with 18 FADs. The firm will pay skipper i according to the amount obtained by vessel i would receive if it recovered 2 more FADs. If vessel i is reassigned with the same or more FADs than it initially had, it will be paid according to the number of reassigned FADs.

In Proposition 2 of the Appendix, we theoretically study this mechanism. Here we discuss the practical implications of Proposition 2. According to part (a), we know that there is a BNE when every skipper decides not to share its FADs. The same applies to Mechanism 1. Per part (c), there is also a BNE when every skipper shares its FADs and the firm reorganizes all the FADs optimally. Further, the utility of the firm and each skipper under part (c) is greater than or equal to when no vessels share FADs.

Next we asked, when is the BNE of part (c) different from that of part (a)? We also asked the extent of these differences. In Example 3 of the Appendix, both BNE are essentially the same. Thus, from a theoretical point of view, the answer to our question is that it depends on the characteristics of the problem. We then offered (in the next section) a practical answer to both questions. After developing simulations based on real data, our results showed that, in all cases studied, the BNE of part (*c*) was different from the one of part (*a*). Additionally, both were quite different in terms of utility obtained by the skippers and the firm. In this case, the firm clearly benefits more than the skippers.

Part (*b*) of Proposition 2 says the following: Suppose that skipper *i* decides between sharing or not sharing its FADs. Independent of the position of its FADs or the decision taken by other skippers, the expected utility obtained sharing its FADs is never smaller than the expected utility obtained not sharing its FADs. This means that the Bayesian Nash equilibria we should observe in practice is the one in which every skipper shares its FADs. Thus, with Mechanism 2, every skipper has incentives to share its FADs, and this mechanism is also suitable for the firm.

4.5 Data and results

In this section we design an experiment that, based on data from FADs movements, assesses the theoretical propositions made in the previous chapter. It is worth recalling that we used real-data from different tuna fishing companies. To test our model exclusively for scientific purposes, Marine Instruments provided us with anonymous real-data from several tuna vessels fishing in the FAO capture zone no. 57 (Eastern Indian Ocean) from April 9 to April 23, 2017. Specifically this experiment was based on a tuna company composed of 3 vessels (i.e., 3 skippers) with 20 FADs per vessel.

This information was obtained randomly using the MSB software, a platform for receiving and visualizing buoy data, from Marine Instruments. We performed 10 measurements in each experiment, varying the positions of the FADs and the vessels, to obtain representative mean values for each case study. We suppose that tuna vessels navigate at 15 knots and, for simplicity, the expected average of tuna by FAD is 6.1 tons, with $29 per nautical mile the cost of fuel at this speed. All these working conditions are represented in Table 4.1 and were obtained from Marine's historical records for vessels working in this area during the last decade:

Table 4.1: Experiment assumptions

Description	**Value**
Number of vessels	3
FADs per vessel	20
Vessel speed	15 knots
Fishing time	3 hours
Tons beneath each FAD	6.1
Cost per ton	$1,400
Fuel cost per mile	$29
Skipper benefit	10%

It should also be noted that, for correct interpretation of the results, all skippers have variable benefits that depend on the quantity of fish they catch. While this value may be different from one company to another, we have supposed an average benefit of about 10% of the total amount of tuna fished, which is based on the average tuna stock price. Although the companies did not give us this information, they confirmed that the supposed percentage is a reasonable value, and this percentage does not affect the theory we want to prove. For simplicity, we paid no attention to the firm's fixed expenditures, such as crew costs, supplies, fishing licenses, etc.

Considering these conditions and following the same structure as in the previous theoretical section, a total of three different scenarios were considered. The first scenario (Table 4.2) describes the current situation where the skippers do not share their FADs. In the second scenario (Table 4.3) the three skippers share the FADs without compensation. Finally, in the last scenario (Table 4.4), the same situation is proposed but with compensation for the skippers to share. Next, each of these three situations is analyzed in detail.

In the first scenario we assumed that the skippers did not share their FADs. In these conditions, therefore, each skipper only knows the position of their own FADs. The results obtained are shown in Table 4.2, where we can observe the money earned by each skipper (vessel) and the money earned by the firm (owner of the three tuna vessels) within the conditions (tons per FAD, cost per ton, fuel cost, etc.) previously illustrated in Table 4.1. To obtain a representative average value (Avg.), we have repeated each simulation 10 times representing different FADs situations. It is worth recalling that we assume the same quantity of fish beneath of each FAD. Therefore, the expected earned money for each skipper is the same for each simulation but it is not for the firm, because totals also depend on how many miles the vessels navigate, and firm benefits depend not only on the amount of tuna captured, but also on the fuel spent; the more miles traveled, the fewer benefits for the firm.

Table 4.2: Current way of working: Vessels do not share their FADs

	Skipper 1	**Skipper 2**	**Skipper 3**	**Ship owner**
1	$ 16,969	$ 16,969	$ 16,969	$ 248,267
2	$ 16,969	$ 16,969	$ 16,969	$ 247,448
3	$ 16,969	$ 16,969	$ 16,969	$ 244,741
4	$ 16,969	$ 16,969	$ 16,969	$ 255,185
5	$ 16,969	$ 16,969	$ 16,969	$ 255,032
6	$ 16,969	$ 16,969	$ 16,969	$ 256,515
7	$ 16,969	$ 16,969	$ 16,969	$ 258,642
8	$ 16,969	$ 16,969	$ 16,969	$ 255,185
9	$ 16,969	$ 16,969	$ 16,969	$ 244,166
10	$ 16,969	$ 16,969	$ 16,969	$ 251,943
Avg.	$ 16,969	$ 16,969	$ 16,969	$ 251,712

In the second scenario (Table 4.3) we assume that the three skippers (vessels) agree to share their FADs, so the firm makes an optimal distribution of the FADs and assigns them in a smart way from the office to the vessels. This means that sometimes one vessel can have 20 FADs, sometimes more and sometimes less. Table 4.3 shows these results, and we can observe that skippers 1 and 2 achieve greater benefits than skipper 3 because, on average, they had more FADs during the simulations.

The firm in this scenario would obtain important benefits because of the fuel saved by this smart distribution of FADs, reaching 8.5% improvement compared to the previous scenario. However, the total benefits of the skippers does not change. Skipper 1 improves 4.3%, skipper 2 improves 0.8% but skipper 3 decreases 5.1% (compared to Table 4.1, which reflects current fishing methods). This results confirm the theoretical results we have seen in the previous section. Skippers do not have not incentive to share their FADs with other vessels because the expected benefit of a skipper when sharing their FADs, is the same that when no sharing the FADs. Thus, as there is no expectation of improvement, it seems very likely that the skippers would not want to take risks and continue working only with their own FADs. Seen from the global point of view of the company (and shareholders), however, the best scenario would involve sharing. Our empirical results corroborate the theoretical assumptions described above and help explain why many tuna vessels work alone. Nevertheless, this mechanism is not the more suitable for the firm.

Table 4.3: Mechanism 1: Reassigning FADs without compensation

	Skipper 1	**Skipper 2**	**Skipper 3**	**Ship owner**
1	$ 17,818	$ 16,121	$ 16,970	$ 281,716
2	$ 17,818	1 $ 6,970	$ 16,121	$ 266,843
3	$ 19,515	$ 15,273	$ 16,121	$ 269,936
4	$ 16,970	$ 16,970	$ 16,970	$ 276,557
5	$ 17,818	$ 16,121	$ 16,970	$ 285,337
6	$ 16,970	$ 18,667	$ 15,273	$ 258,980
7	$ 17,309	$ 18,723	$ 14,877	$ 269,586
8	$ 17,164	$ 19,063	$ 14,683	$ 270,121
9	$ 19,515	$ 16,121	$ 15,273	$ 279,064
10	$ 16,121	$ 16,970	$ 17,818	$ 272,243
Avg.	$ 17,702	$ 17,100	$ 16,107	$ 273,038
Diff	4.3%	0.8%	-5.1%	8.5%

In the third scenario the firm changes its strategy of incentives for the skippers, as shown in Proposition 2 (see Mechanism 2: Reassigning FADs with compensation in Section 3). In this scenario, the firm guarantees pay to vessels that share FADs at least according to the number of FADs the vessel initially had. In other words, when a skipper has fewer FADs assigned than the average, he or she is automatically compensated by the firm. For example, when a skipper has 2 FADs fewer than the current situation (i.e., 20), the company will still pay for 20 FADs, so there is not any risk for the skipper. But, when the skipper has more FADs assigned than the average (for instance, 21), he or she will keep them, fishing more, and the quantity of fish expected for each vessel will be more than in the first scenario. With this policy of incentives, it seems logical to expect the bosses to be encouraged to collaborate since everybody wins. In fact, as was theoretically proved in Proposition 2, every skipper has incentives to share its FADs; therefore, the Bayesian Nash equilibria we should observe in practice with real-data is the one in which every vessel shares its FADs.

The results are shown in Table 4.4, where we can clearly see that each skipper enjoys more benefits than in the first scenario where did not share. Further, the firm continues earning more than the first scenario, although less than the second, as it was expected. In this way, our empirical findings complement the previous theoretical section. While, theoretically, we could only predict that it was favorable to share FADs, the simulations performed not only confirm this but also verify that both the company and the skippers get more benefit with this new procedure. In fact, with our data, we can also estimate how much more they will earn on average over time. We can assure, therefore, that this mechanism is suitable for the firm.

Table 4.4: Mechanism 2: Reassigning FADs with compensation

	Skipper 1	**Skipper 2**	**Skipper 3**	**Ship owner**
1	$ 17,818	$ 16,969	$ 16,969	$ 280,867
2	$ 17,818	$ 16,969	$ 16,969	$ 265,994
3	$ 19,515	$ 16,969	$ 16,969	$ 267,390
4	$ 16,969	$ 16,969	$ 16,969	$ 276,557
5	$ 17,818	$ 16,969	$ 16,969	$ 284,488
6	$ 16,969	$ 18,666	$ 16,969	$ 257,283
7	$ 17,309	$ 18,723	$ 16,969	$ 267,492
8	$ 17,163	$ 19,062	$ 16,969	$ 267,834
9	$ 19,515	$ 16,969	$ 16,969	$ 276,518
10	$ 16,969	$ 16,969	$ 17,818	$ 271,394
Avg.	$ 17,786	$ 17,524	$ 17,054	$ 271,582
Diff	4.8%	3.3%	0.5%	7.9%

We used the nearest neighbor strategy for recovering FADs during the simulations. This means that FAD distribution was based on assigning FADs closer to each tuna vessel. This is a quick and sound distribution method commonly used by the tuna industry at present, but it is far from being optimal. The results could be further improved if this recovery strategy was changed to adapt to the dynamic nature of drifting FADs, as is shown in Groba et al (2018). In this case, it was proved that the quantity of miles traveled could be reduced by 21.4% in the case of 3 veseels and 20 FADs per vessel in comparison with the NN strategy, as we use in our approach. It not only indicates that the firm is going to earn more due to the route optimization, but also that fishing time will be reduced, so skippers can fish the same quantity in less time and still gain all the associated economic and environmental implications.

4.6 Conclusions

Based on data from different groups of tuna vessels retrieving their FADs in the Indian Ocean during 2017, this paper proposes a new, coordinated way of working for the tuna fishing companies related to FAD collection. Situated within well-known game theory, our findings reflect the value of sharing FADs. We demonstrate that, with the correct incentives, there is a situation in which all stakeholders –company, skipper, and environment– obtain better results. Further, global economic profits are realized for the fleet and company, and CO_2 emissions are reduced.

From a scholarly perspective, our work provides empirical evaluation using real-data and supports and applies the adequacy of the proposed theoretical model –a non cooperative game with incomplete information following the model of Aumann considering Bayesian Nash equilibria– to a complex, real-world situation. We assumed two different situations: 1) reassigning FADs without compensation and 2) reassigning FADs with compensation. While in the first situation our results only corroborate theoretical assumptions (and explain why tuna vessels work alone), the empirical portion of the second situation complements the previous theoretical section. While, theoretically, we could only predict that it was favorable to share FADs, the simulations performed not only allow us to confirm this and verify that both the company and the skippers get more benefit with this new procedure. Additionally, with our data, we can also estimate how much more they each will earn on average over time. We can assure, therefore, that this mechanism is suitable for the firm.

As the firm will enjoy savings due to the route optimization, tuna vessels will reduce their fishing time and fuel consumed. In addition, fuel reduction presents another important advantage: increased storage capacity. This paper opens up, therefore, a set of possibilities for a wide range of real-world problems.

Similarly, from a policy-maker perspective, our work addresses a new, more-efficient way to work with increasing FAD regulations regarding the number of FADs per PS. While, these regulations were first introduced in the Indian Ocean by the Indian Ocean Tuna Commission (IOTC) with Resolution 17/08, it is very likely that these regulatory restrictions will soon extend to the rest of the oceans by the ICCAT and the IATTC. It is mandatory, therefore, for the tuna fishing industry to optimize the use of FADs. As it seems clear that this number will be drastically reduced in the next few years, there is no other way but to use them as efficiently as possible.

From an environmental perspective, our proposal would directly reduce the total current CO_2 emissions. This is a significant improvement, as climate change is one of the main problems facing humanity today (Howard-Grenville et al, 2014). In addition, the development of more sustainable fishing methods using FADs may be possible because of our research.

Acknowledgements

The authors wish to thank Marine Instruments for the buoy data. The authors are partially supported by research grants ECO2016-76625-R and ECO2017-82241-R from the Spanish Ministry of Science and Competitiveness, GRC 2014/021 and GRC 2015/014 from "Xunta de Galicia".

4.A Appendix

We now introduce some well known concepts of non cooperative game theory. We refer to Zamir (2013) for a detailed discussion of such concepts.

An **Aumann model of incomplete information** (Aumann (1976)) is a tuple

$$\left(I, X, (\pi_i)_{i \in I}, P\right)$$

where I is the set of agents; X is the set of states of the world; for each $i \in I$, π_i is a partition of X; and P is a probability distribution over X (called common prior).

Given $x \in X$ and $i \in I$ we denote by $\pi_i(x)$ the element of π_i to which x belongs to.

The interpretation is as follows. There is a possible set of states of the world (X) and a probability distribution (P) over X known by all agents. An element $x \in X$ is randomly selected according with P. Each agent $i \in I$ has different information about such element, which is given by π_i. We assume that agent i knows that an element of $\pi_i(x)$ has happened, but he/she can not distinguishes among the elements of $\pi_i(x)$.

The Harsanyi's model of incomplete information (Harsanyi (1967) is more popular than the Aumann's model of incomplete information. In this paper we use Aumann's model because it fits better with the problem we are studying.

For each state of the world $x \in X$ we consider the classical non-cooperative game $\Gamma^x = \left(I, (A_i^x)_{i \in I}, (u_i^x)_{i \in I}\right)$ played at this state. For each agent $i \in I$, A_i^x denotes the set of pure actions that agent i can take when the state of the world is x. We assume that $A_i^x = A_i^{x'}$ when $\pi_i(x) = \pi_i(x')$. Besides $u_i^x : \times_{i \in I} A_i^x \to \mathbb{R}$ represents the utility of agent i.

An strategy for agent i is a mapping σ_i assigning to each state of the world $x \in X$ an action $\sigma_i(x) \in A_i^x$ such that $\sigma_i(x) = \sigma_i(x')$ when $\pi_i(x) = \pi_i(x')$. We denote by Σ_i the set of all strategies of agent i.

A Bayesian game on X is a triple $\left(I, (\Sigma_i)_{i \in I}, (u_i)_{i \in I}\right)$ where for each $\sigma = (\sigma_i)_{i \in I}$ and each $i \in N$

$$u_i(\sigma) = \int_X u_i^x \left((\sigma_i(x))_{i \in I}\right) dP$$

A **Bayesian Nash equilibria (briefly BNE)** is a tuple $\sigma = (\sigma_i)_{i \in I}$ such that for for each $i \in I$ and each $\sigma_i' \in \Sigma_i$ we have that $u_i(\sigma) \geq u_i(\sigma \backslash \sigma_i')$ where $\sigma \backslash \sigma_i'$ is the combination of strategies where agent i plays σ_i' and each agent $j \in I \backslash \{i\}$ plays σ_j.

Intuitively, in a BNE at each stage of the world x each agent i is playing a best reply against the strategies of the other agents. Thus, a BNE is an extension of the Nash equilibria (Nash (1951)) to this setting.

4.A.1 Mechanism 1. Reassigning FADs without compensation

We consider the Aumann model of incomplete information $\left(I, X, (\pi_i)_{i \in I}, P\right)$ defined as follows.

- $I = \{f, 1, ..., n\}$ where f is the firm and i denotes vessel i for each $i = 1, ..., n$.

  In our simulations we will take 3 vessels. Namely, $n = 3$.

- X is the set of possible locations of the $\tau = \sum_{i=1}^{n} n_i$ FADs assigned to the vessels. Namely $X = Z^{\tau}$ where Z denotes the set of places where a FAD can be located. We assume that coordinates 1 to n_1 from Z^{τ} refer to the position of the FADs assigned to vessel 1. Coordinates n_1+1 to n_1+n_2 from Z^{τ} refer to the position of the FADs assigned to vessel 2 and so on. A generic element of X will be denoted as $x = (x_j)_{j=1}^{\tau}$.

  In our simulations we take Z as the Indic Ocean. Besides each vessel will have 20 FADs ($n_i = 20$ for all $i \in N$) and hence $\tau = 60$.

- $(\pi_i)_{i\in I}$ model the situation where each vessel only knows the position of its FADs and the firm knows the position of all FADs.

  Given $i \in N$ and $x, x' \in X$ we have that $\pi_i(x) = \pi_i(x')$ if and only if the position of the FADs assigned to vessel i in x and x' are the same. Namely, for each $j = \sum_{k=1}^{i-1} n_k + 1, ..., \sum_{k=1}^{i} n_k$ we have that $x_j = x'_j$.

  For each $x \in X$, $\pi_f(x) = \{x\}$.

- P is a probability distribution over X. We do not consider a specific distribution for P because our theoretical results hold for any P.

The non-cooperative game $\Gamma^x = \left(I, (A_i^x)_{i\in I}, (u_i^x)_{i\in I}\right)$ we consider is defined for modelling the following situation. Each vessel, independently, decides if it share its FADs with other vessels. If a vessel says no, then such vessel remains with the same FADs. Among the vessels that say yes, the firm reassign the FADs of such vessels among themselves. We now formalize this idea.

- I as above.

- $(A_i^x)_{i\in I}$. For each $i \in N$, $A_i^x = \{YES, NO\}$.

  Let $N^{x,YES}$ the set of vessels that says YES. Let

$$B^{x,YES} = \bigcup_{i\in N^{x,YES}} \bigcup_{k=1}^{n_i} b_i^k$$

  be the set of all FADs assigned initially to vessels that said YES.

  A_f^x is the set of all possible reallocations of the FADs of $B^{x,YES}$ among agents in $N^{x,YES}$. Namely,

$$A_f^x = \left\{ \begin{array}{c} (B_i)_{i\in N^{x,YES}} : \text{for each } i \in N^{x,YES},\ \varnothing \subset B_i \subset B^{x,YES}, \\ \bigcup_{i\in N^{x,YES}} B_i = B^{x,YES} \text{ and} \\ B_i \cap B_j = \varnothing \text{ for each } i,j \in N^{x,YES},\ i \neq j. \end{array} \right\}$$

- $(u_i^x)_{i\in I}$. Let $(a_i^x)_{i\in I} \in \times_{i\in I} A_i^x$.

  Let $j \in N$ be a vessel that said NO. Then, the vessel continue with the same FADs, b_j. Hence its utility is $u_j\left((a_i^x)_{i\in I}\right) = pn_jq$.

Let $j \in N$ be a vessel that said YES. Then, the vessel has a new set of FADs, B_j. Hence its utility is $u_j\left((a_i^x)_{i\in I}\right) = p\left|B_j\right|q$ where $\left|B_j\right|$ denotes the number of FADs in B_j.

Finally, the utility of the firm is

$$u_f\left((a_i^x)_{i\in I}\right) = (p_f - p)\sum_{i=1}^{n} n_j q - c\sum_{i\in N\setminus N^{x,YES}} d(b_i) - c\sum_{i\in N^{x,YES}} d(B_i)$$

The utility of the firm has three parts. The first one, $(p_f - p)\sum_{i=1}^{n} n_j q$, corresponds to the benefits of selling the fish. This part is independent of the actions taking by the vessels. The second one, $-c\sum_{i\in N\setminus N^{x,YES}} d(b_i)$, corresponds to the cost of the fuel of the vessels that did not share its FADs. This part depends on the actions of the vessels but not on the action of the firm. The third one, $-c\sum_{i\in N^{x,YES}} d(B_i)$, corresponds to the cost of the fuel of the vessels that shared its FADs. This part depends on the actions of the vessels and on the action of the firm.

We now make a theoretical analysis of the Bayesian game $\left(I, (\Sigma_i)_{i\in I}, (u_i)_{i\in I}\right)$ associated to this case.

Proposition 1. Let $\left(I, X, (\pi_i)_{i\in I}, P\right)$ be the Aumann model of incomplete information defined as above.

(a) Let $\sigma = (\sigma_i)_{i\in I}$ be such that for each $i \in N$ and for each $x \in X$, $\sigma_i(x) = NO$. Then, σ is a BNE of $\left(I, (\Sigma_i)_{i\in I}, (u_i)_{i\in I}\right)$ and for each $i \in N$, $u_i(\sigma) = pn_i q$.

(b) Let $\sigma = (\sigma_i)_{i\in I}$ be a BNE of $\left(I, (\Sigma_i)_{i\in I}, (u_i)_{i\in I}\right)$. Then, for each $i \in N$, $u_i(\sigma) = pn_i q$.

Proof of Proposition 1. We first note that for each $i \in I$ and each $\sigma = (\sigma_i)_{i\in I}$ we have that

$$u_i(\sigma) = \int_X u_i^x\left((\sigma_i(x))_{i\in I}\right) dP = \sum_{X_i\in\pi_i}\int_{X_i} u_i^x\left((\sigma_i(x))_{i\in I}\right) dP$$

and $\sigma_i(x) = \sigma_i(x')$ for all $x, x' \in X_i$.

(a) We have to prove that for each $i \in I$ and each $\sigma_i' \in \Sigma_i$, we have that $u_i(\sigma) \geq u_i(\sigma\backslash\sigma_i')$.

Let $i = f$. Since all vessels are saying NO, firm has nothing to do. Then, for each $\sigma_f' \in \Sigma_f$ we have that $u_f(\sigma) = u_f\left(\sigma\backslash\sigma_f'\right)$.

Let $i \in N$ and $\sigma_i' \in \Sigma_i$. Thus,

$$u_i\left(\sigma\backslash\sigma_i'\right) = \sum_{X_i\in\pi_i}\int_{X_i} u_i^x\left(\sigma_i'(x), (\sigma_j(x))_{j\in I\setminus\{i\}}\right) dP.$$

Let $X_i \in \pi_i$ be such that $\sigma_i'(x) = NO$ for each $x \in X_i$. Since $\sigma_i(x) = NO$ for each $x \in X_i$ we have that

$$\int_{X_i} u_i^x\left(\sigma_i'(x), (\sigma_j(x))_{j\in I\setminus\{i\}}\right) dP = \int_{X_i} u_i^x\left((\sigma_j(x))_{j\in I}\right) dP.$$

Let $X_i \in \pi_i$ be such that $\sigma_i'(x) = YES$ for each $x \in X_i$. Thus, $N^{x,YES} = \{i\}$ and $B^{x,YES} = \bigcup_{k=1}^{n_i} b_i^k$. Hence A_f^x, the set of all possible reallocations of the FADs of $B^{x,YES}$ among agents in $N^{x,YES}$ has a unique element, namely, to assign all the FADs of vessel i to vessel i. Thus,

$$\int_{X_i} u_i^x\left(\sigma_i'(x), (\sigma_j(x))_{j\in I\setminus\{i\}}\right) dP = \int_{X_i} u_i^x\left((\sigma_j(x))_{j\in I}\right) dP.$$

Hence,

$$u_i\left(\sigma\setminus\sigma_i'\right) = \sum_{X_i\in\pi_i} \int_{X_i} u_i^x\left((\sigma_j(x))_{j\in I}\right) dP = u_i(\sigma).$$

(b) We first prove a couple of statements that will be used in the proof of this part.

Statement 1. Let $\sigma = (\sigma_i)_{i\in I}$ be a BNE of $\left(I, (\Sigma_i)_{i\in I}, (u_i)_{i\in I}\right)$. Then, there exists $X' \subset X$ such that $\int_{X'} dP = 1$ and for each $x \in X'$, $\sigma_f(x) = (B_i^*)_{i\in N^{x,YES}}$ where

$$\sum_{i\in N^{x,YES}} d(B_i^*) = \min\left\{\sum_{i\in N^{x,YES}} d(B_i) : (B_i)_{i\in N^{x,YES}} \in B^{x,YES}\right\}.$$

Proof of Statement 1. For each $x \in X$ we denote $\sigma_f(x) = (B_i)_{i\in N^{x,YES}}$. Besides, we define $X'' = \left\{x \in X : \sigma_f(x) \neq (B_i^*)_{i\in N^{x,YES}}\right\}$.

Suppose that the statement does not hold. Then, $\int_{X''} dP > 0$.

We now define σ_f' such that $\sigma_f'(x) = (B_i^*)_{i\in N^{x,YES}}$ for all $x \in X$. Then

$$\begin{aligned} u_f\left(\sigma\setminus\sigma_f'\right) &= \int_X u_f^x\left(\sigma_f'(x), (\sigma_j(x))_{j\in I\setminus\{f\}}\right) dP \\ &= \int_{X\setminus X''} u_f^x\left(\sigma_f'(x), (\sigma_j(x))_{j\in I\setminus\{f\}}\right) dP \\ &\quad + \int_{X''} u_f^x\left(\sigma_f'(x), (\sigma_j(x))_{j\in I\setminus\{f\}}\right) dP. \end{aligned}$$

Since $\sigma_f'(x) = \sigma_f(x)$ for all $x \in X\setminus X''$ we have that

$$\int_{X\setminus X''} u_f^x\left(\sigma_f'(x), (\sigma_j(x))_{j\in I\setminus\{f\}}\right) dP = \int_{X\setminus X''} u_f^x\left((\sigma_j(x))_{j\in I}\right) dP.$$

Besides,

$$\begin{aligned} &\int_{X''} u_f^x\left(\sigma_f'(x), (\sigma_j(x))_{j\in I\setminus\{f\}}\right) dP \\ = &\int_{X''}\left[(p_f - p)\sum_{i=1}^n n_j q - c\sum_{i\in N\setminus N^{x,YES}} d(b_i) - c\sum_{i\in N^{x,YES}} d(B_i^*)\right] dP \\ > &\int_{X''}\left[(p_f - p)\sum_{i=1}^n n_j q - c\sum_{i\in N\setminus N^{x,YES}} d(b_i) - c\sum_{i\in N^{x,YES}} d(B_i)\right] dP \\ = &\int_{X''} u_f^x\left((\sigma_j(x))_{j\in I}\right) dP. \end{aligned}$$

Thus,

$$\begin{aligned} u_f\left(\sigma\backslash\sigma_f'\right) &> \int_{X\backslash X''} u_f^x\left(\left(\sigma_j(x)\right)_{j\in I}\right)dP + \int_{X''} u_f^x\left(\left(\sigma_j(x)\right)_{j\in I}\right)dP \\ &= \int_X u_f^x\left(\left(\sigma_j(x)\right)_{j\in I}\right)dP = u_f(\sigma), \end{aligned}$$

which contradicts that σ is a BNE. ■

<u>Statement 2</u>. Let $\sigma=(\sigma_i)_{i\in I}$ be a BNE of $\left(I,(\Sigma_i)_{i\in I},(u_i)_{i\in I}\right)$. For each $x\in X$ and each $i\in N$ such that $\int_{\pi_i(x)}dP>0$ we have that

$$\int_{x'\in\pi_i(x)} u_i^{x'}\left(\left(\sigma_i(x')\right)_{i\in I}\right)dP \geq pn_iq\int_{\pi_i(x)}dP.$$

<u>Proof of Statement 2</u>. Let $x\in X$ and $i\in N$ such that $\int_{\pi_i(x)}dP>0$ and $\sigma_i(x)=NO$. Then, vessel i receives its initial FADs and hence

$$\int_{x'\in\pi_i(x)} u_i^{x'}\left(\left(\sigma_i(x')\right)_{i\in I}\right)dP = \int_{x'\in\pi_i(x)} pn_iq\,dP = pn_iq\int_{\pi_i(x)}dP.$$

Let $x\in X$ and $i\in N$ such that $\int_{\pi_i(x)}dP>0$ and $\sigma_i(x)=YES$. Suppose not. Then,

$$\int_{x'\in\pi_i(x)} u_i^{x'}\left(\left(\sigma_i(x')\right)_{i\in I}\right)dP < pn_iq\int_{\pi_i(x)}dP.$$

Let σ_i' be such that $\sigma_i'(x')=NO$ when $x'\in\pi_i(x)$ and $\sigma_i'(x')=\sigma_i(x')$ otherwise. Now,

$$\begin{aligned} u_i\left(\sigma\backslash\sigma_i'\right) &= \sum_{X_i\in\pi_i}\int_{x'\in X_i} u_i^{x'}\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)dP \\ &= \sum_{X_i\in\pi_i\backslash\pi_i(x)}\int_{x'\in X_i} u_i^{x'}\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)dP \\ &\quad+\int_{x'\in\pi_i(x)} u_i^{x'}\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)dP \end{aligned}$$

Since $\sigma_i'(x')=\sigma_i(x')$ when $x'\in X_i\in\pi_i\backslash\pi_i(x)$ we have that

$$\sum_{X_i\in\pi_i\backslash\pi_i(x)}\int_{x'\in X_i} u_i^{x'}\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)dP = \sum_{X_i\in\pi_i\backslash\pi_i(x)}\int_{x'\in X_i} u_i^{x'}\left(\left(\sigma_j(x')\right)_{j\in I}\right)dP.$$

Since $u_i^x\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)=pn_iq$ when $x'\in\pi_i(x)$ we have that

$$\begin{aligned} \int_{x'\in\pi_i(x)} u_i^{x'}\left(\sigma_i'(x'),\left(\sigma_j(x')\right)_{j\in I\backslash\{i\}}\right)dP &= pn_iq\int_{\pi_i(x)}dP \\ &> \int_{x'\in\pi_i(x)} u_i^{x'}\left(\left(\sigma_i(x')\right)_{i\in I}\right)dP. \end{aligned}$$

Thus,

$$\begin{aligned} u_i\left(\sigma\backslash\sigma_i'\right) &> \sum_{X_i\in\pi_i\backslash\pi_i(x)}\int_{x'\in X_i} u_i^{x'}\left(\left(\sigma_j(x')\right)_{j\in I}\right)dP + \int_{x'\in\pi_i(x)} u_i^{x'}\left(\left(\sigma_i(x')\right)_{i\in I}\right)dP \\ &= u_i(\sigma), \end{aligned}$$

which contradicts that σ is a BNE. ■

We now prove (b). We know that

$$u_i(\sigma) = \sum_{X_i\in\pi_i}\int_{X_i} u_i^x\left((\sigma_i(x))_{i\in I}\right)dP = \sum_{X_i\in\pi_i:\int_{X_i}dP>0}\int_{X_i} u_i^x\left((\sigma_i(x))_{i\in I}\right)dP$$

By statement 2,

$$\sum_{X_i\in\pi_i:\int_{X_i}dP>0}\int_{X_i} u_i^x\left((\sigma_i(x))_{i\in I}\right)dP \geq pn_iq \sum_{X_i\in\pi_i:\int_{X_i}dP>0}\int_{X_i} dP.$$

Since P is a probability, $\int_X dP = 1$. Then,

$$pn_iq \sum_{X_i\in\pi_i:\int_{X_i}dP>0}\int_{X_i} dP = pn_iq\int_X dP = pn_iq.$$

Besides,

$$\begin{aligned}
\sum_{i=1}^n u_i(\sigma) &= \sum_{i=1}^n \int_X u_i^x\left((\sigma_i(x))_{i\in I}\right)dP = \int_X \sum_{i=1}^n u_i^x\left((\sigma_i(x))_{i\in I}\right)dP \\
&= \int_X \left[\sum_{i\in N\setminus N^{x,YES}} pn_iq + \sum_{i\in N^{x,YES}} p\left|B_i\right|q\right]dP \\
&= pq\int_X \left[\sum_{i\in N\setminus N^{x,YES}} n_i + \sum_{i\in N^{x,YES}} n_i\right]dP \\
&= pq\int_X \left[\sum_{i\in N} n_i\right]dP = pq\sum_{i\in N} n_i \\
&= \sum_{i\in N} pn_iq.
\end{aligned}$$

Since $u_i(\sigma) \geq pn_iq$ for each $i \in N$ and $\sum_{i=1}^{n} u_i(\sigma) = \sum_{i\in N} pn_iq$ we deduce that $u_i(\sigma) = pn_iq$ for each $i \in N$.

Proposition 1 says nothing about the utility obtained by the firm. Thus, a natural question that arises is the following: is it possible to find BNE where some vessels share its FADs? Notice that if the answer is YES, then the firm can improve its utility by the fuel's savings.

Next examples show that the answer depends on P and the location of the FADs.

Example 1. Consider the case where we have two vessels ($I = \{f, a, b\}$) and each vessels has two FADs. Besides every vessel knows the location of every FAD. Namely, P assign probability 1 to element $x = \left(b_a^1, b_a^2, b_b^1, b_b^2\right)$ and zero to the rest of elements of X. The distances between the FADs and the vessels are the following:

distances	1	2	b_a^1	b_a^2	b_b^1
2	50				
b_a^1	5	45			
b_a^2	35	15	30		
b_b^1	15	35	10	20	
b_b^2	45	5	40	10	30

The distances are computed by assuming that vessels are located in a line. From left to right $1\,[5]\,b_a^1\,[10]\,b_b^1\,[20]\,b_a^2\,[10]\,b_b^2\,[5]\,2$. The distance between vessel 1 and FADs b_a^1 is 5; the distance between FADs b_a^1 and b_b^1 is 10 and so on.

Let σ be the BNE where each vessel says NO. Then, each vessels recover its FADs. Vessel a moves to FAD b_a^2 (distance 5), next to FAD b_a^2 (distance 30) and then back (35). The total distance traveled is 70. Similarly, the distance traveled by vessel b is also 70. Then, the utility of each vessel is $2pq$ and the utility of the firm is $(p_f - p)4q - c140$.

Let σ be such that each vessel says YES and the firm assign FADs b_a^1 and b_b^1 to vessel a and FADs b_a^2 and b_b^2 to vessel b. It is easy to see that σ is a BNE. Besides the utility of the firm is $(p_f - p)4q - c60$. Thus, in this BNE the firm can improve with respect to the initial situation.

Example 2. Consider the same case as in Example 1 but now the distances between the FADs and the vessels are the following:

distances	1	2	b_a^1	b_a^2	b_b^1
2	140				
b_a^1	5	135			
b_a^2	105	15	120		
b_b^1	125	35	100	20	
b_b^2	135	5	110	10	30

The distances are computed by assuming that vessels are located in a line. From left to right $1\,[5]\,b_a^1\,[100]\,b_b^1\,[20]\,b_a^2\,[10]\,b_b^2\,[5]\,2$.

In this example the unique BNE is the one where each vessel says NO. Notice that if both vessels say YES then the firm assign FAD b_a^1 to vessel a and the other FADs to vessel b. Thus, vessel a is better saying NO than saying YES.

4.A.2 Mechanism 2. Reassigning FADs with compensation

We now introduce the theoretical model for analyzing this case. The Aumann model of incomplete information $\left(I, X, (\pi_i)_{i\in I}, P\right)$ associated to this case is the same as above.

The non-cooperative game $\Gamma^x = \left(I, (A_i^x)_{i\in I}, (u_i^x)_{i\in I}\right)$ we consider is defined bas follows. $(A_i^x)_{i\in I}$ is the same as in Case 1. Nevertheless $(u_i^x)_{i\in I}$ will be modified in order to consider the compensation that firm give to vessels that share its FADs. Let $(a_i^x)_{i\in I} \in \times_{i\in I} A_i^x$.

Let $j \in N$ be a vessel that said NO (namely $a_j^x = NO$). Then, the vessel continue with the same FADs, b_i. Hence its utility is $u_j\left((a_i^x)_{i\in I}\right) = pn_jq$. This utility is the same as in Mechanism 1.

Let $j \in N$ be a vessel that said YES (namely $a_j^x = YES$). Then, the vessel has a new set of assigned FADs, B_j. Hence its utility is

$$u_j\left((a_i^x)_{i\in I}\right) = p\max\left\{\left|B_j\right|, n_j\right\} q$$

where $\left|B_j\right|$ denotes the number of FADs in B_j. Notice that if vessel j receives at least n_j FADs, then it will be paid according with the FADs received. If vessel j receives less than n_j, then it will be paid as if the vessel receives n_j FADs. In this part appears clearly the new incentive mechanism.

Finally, the utility of the firm $u_f\left((a_i^x)_{i\in I}\right)$ is given by

$$(p_f-p)\sum_{i=1}^{n} n_j q - c\sum_{i\in N\setminus N^{x,YES}} d(b_i) - c\sum_{i\in N^{x,YES}} d(B_i) - \sum_{i\in N^{x,YES},|B_i|<n_i} p\,(n_i-|B_i|)\,q$$

The utility of the firm has four parts. The first one, $(p_f-p)\sum_{i=1}^{n} n_j q$, the second one, $-c\sum_{i\in N\setminus N^{x,YES}} d(b_i)$, and the third one, $-c\sum_{i\in N^{x,YES}} d(B_i)$, are the same as in the previous case. In this case it appears a fourth one,

$$-\sum_{i\in N^{x,YES},|B_i|<n_i} p\,(n_i-|B_i|)\,q,$$

where it appears the compensation that the firm gives to the vessels that say YES and receive less FADs than initially.

We now make a theoretical analysis of the Bayesian game $\left(I,(\Sigma_i)_{i\in I},(u_i)_{i\in I}\right)$ associated to this case.

Proposition 2. Let $\left(I,X,(\pi_i)_{i\in I},P\right)$ be the Aumann model of incomplete information defined as above.

(a) Let $\sigma^{NO}=\left(\sigma_i^{NO}\right)_{i\in I}$ be such that for each $i\in N$ and for each $x\in X$, $\sigma_i^{NO}(x)=NO$. Then, σ^{NO} is a BNE of $\left(I,(\Sigma_i)_{i\in I},(u_i)_{i\in I}\right)$ and for each $i\in N$, $u_i\left(\sigma^{NO}\right)=pn_iq$.

(b) Let $i\in N$ and $x\in X$. We define $a_i^{x,YES}=YES$ and $a_i^{x,NO}=NO$. For each $\left(a_j^x\right)_{j\in I}\in\times_{i\in I}A_i^x$ we have that

$$\int_{x'\in\pi_i(x)} u_i^{x'}\left(a_i^{x,YES},\left(a_j^x\right)_{j\in I\setminus\{i\}}\right)dP \geq \int_{x'\in\pi_i(x)} u_i^{x'}\left(a_i^{x,NO},\left(a_j^x\right)_{j\in I\setminus\{i\}}\right)dP.$$

(c) There exists a BNE $\sigma^{YES}=\left(\sigma_i^{YES}\right)_{i\in I}$ of $\left(I,(\Sigma_i)_{i\in I},(u_i)_{i\in I}\right)$ where for each $i\in N$ and for each $x\in X$, $\sigma_i^{YES}(x)=YES$. Besides, for each $i\in I$, $u_i\left(\sigma^{YES}\right)\geq u_i\left(\sigma^{NO}\right)$.

Proof of Proposition 2. (a) It is similar to the proof of Proposition 1 (a).

(b) We know that for each $x'\in\pi_i(x)$ and each $\left(a_j^x\right)_{j\in I}\in\times_{i\in I}A_i^x$

$$\begin{aligned} u_i^{x'}\left(a_i^{x,NO},\left(a_j^x\right)_{j\in I\setminus\{i\}}\right) &= pn_iq \text{ and}\\ u_i^{x'}\left(a_i^{x,YES},\left(a_j^x\right)_{j\in I\setminus\{i\}}\right) &= p\max\left\{n_i,\left|B_i^{x'}\right|\right\}q \end{aligned}$$

where $B_i^{x'}$ is the set of FADs assigned to vessel i after saying YES. Thus, the result holds trivially.

(c) For each $x\in X$ we take $\sigma_f^{YES}(x)=(B_i^*)_{i\in N^{x,YES}}$ where

$$c\sum_{i\in N^{x,YES}} d(B_i^*)+\sum_{i\in N^{x,YES},|B_i|<n_i} p\,(n_i-|B_i^*|)\,q$$

$$=\min\left\{c\sum_{i\in N^{x,YES}} d(B_i)+\sum_{i\in N^{x,YES},|B_i|<n_i} p\,(n_i-|B_i|)\,q:(B_i)_{i\in N^{x,YES}}\in B^{x,YES}\right\}$$

We first prove that σ^{YES} is a BNE. We need to prove that for each $i \in I$ and each $\sigma_i \in \Sigma_i$ we have that $u_i\left(\sigma^{YES}\right) \geq u_i\left(\sigma^{YES} \backslash \sigma_i\right)$.

Because of the definition of σ_f^{YES} it is clear that for any $\sigma_f \in \Sigma_f$, $u_f\left(\sigma^{YES}\right) \geq u_f\left(\sigma^{YES} \backslash \sigma_f\right)$.

Let $i \in N$ and $\sigma_i \in \Sigma_i$. In the proof of Proposition 1 we have seen that

$$u_i\left(\sigma^{YES}\right) = \sum_{X_i \in \pi_i} \int_{X_i} u_i^x\left(\left(\sigma_i^{YES}(x)\right)_{i \in I}\right) dP.$$

Let $X_i \in \pi_i$ be such that $\sigma_i(x) = YES$ when $x \in X_i$. Then, $\sigma_i^{YES}(x) = \sigma_i(x)$ and hence,

$$\int_{X_i} u_i^x\left(\left(\sigma_i^{YES}(x)\right)_{i \in I}\right) dP = \int_{X_i} u_i^x\left(\sigma_i(x), \left(\sigma_i^{YES}(x)\right)_{i \in I \backslash \{i\}}\right) dP.$$

Let $X_i \in \pi_i$ be such that $\sigma_i(x) = NO$ when $x \in X_i$. By part (b),

$$\int_{X_i} u_i^x\left(\left(\sigma_i^{YES}(x)\right)_{i \in I}\right) dP \geq \int_{X_i} u_i^x\left(\sigma_i(x), \left(\sigma_i^{YES}(x)\right)_{i \in I \backslash \{i\}}\right) dP.$$

Thus,

$$u_i\left(\sigma^{YES}\right) \geq \sum_{X_i \in \pi_i} \int_{X_i} u_i^x\left(\sigma_i(x), \left(\sigma_i^{YES}(x)\right)_{i \in I \backslash \{i\}}\right) dP = u_i\left(\sigma^{YES} \backslash \sigma_i\right).$$

We now prove that for each $i \in I$, $u_i\left(\sigma^{YES}\right) \geq u_i\left(\sigma^{NO}\right)$. Using part (b) it is straightforward to prove that for each $i \in N$, $u_i\left(\sigma^{YES}\right) \geq u_i\left(\sigma^{NO}\right)$.

Since no vessel share its FADs in σ^{NO} we have that for each $x \in X$,

$$u_f^x\left(\sigma^{NO}\right) = (p_f - p) \sum_{i=1}^{n} n_j q - c \sum_{i \in N} d(b_i)$$

and hence

$$u_f\left(\sigma^{NO}\right) = \int_X u_f^x\left(\sigma^{NO}\right) dP = (p_f - p) \sum_{i=1}^{n} n_j q - c \sum_{i \in N} d(b_i).$$

We know that

$$u_f\left(\sigma^{YES}\right) = (p_f - p) \sum_{i=1}^{n} n_j q - c \sum_{i \in N} d(B_i^*) - \sum_{i \in N, |B_i| < n_i} p(n_i - |B_i^*|) q$$

where $\{B_i^*\}_{i=1}^n$ is obtained through the minimization problem defined above.

Thus, for proving that $u_f\left(\sigma^{YES}\right) \geq u_f\left(\sigma^{NO}\right)$ is enough to prove that it exists $\{B_i\}_{i=1}^n$ such that

$$c \sum_{i \in N} d(B_i) + \sum_{i \in N, |B_i| < n_i} p(n_i - |B_i|) q \leq c \sum_{i \in N} d(b_i).$$

If we take $B_i = b_i$ for all $i = 1, ..., n$ we realize that the previous inequality holds. ■

Next example shows that in some cases, the BNE of parts (a) and (c) could be, in a practical way, the same. Nevertheless our simulations based on real-data will show that both BNE could be very different.

Example 3. Consider the same case as in Example 1 but now the distances between the FADs and the vessels are the following:

distances	1	2	b_a^1	b_a^2	b_b^1
2	120				
b_a^1	5	115			
b_a^2	10	110	5		
b_b^1	110	10	105	100	
b_b^2	115	5	105	10	5

The distances are computed by assuming that vessels are located in a line. From left to right $1\,[5]\,b_1^1\,[15]\,b_1^2\,[100]\,b_2^1\,[5]\,b_2^2\,[5]\,2$.

In the BNE described in part (a) each vessels recover its FADs. Then, the utility of each vessel is $2pq$ and the utility of the firm is $(p_f - p)4q - c40$. In the BNE described in part (c) each vessels share its FADs. Then the firm reassign all FADs. But the optimal solution is to assign to each vessel its initial FADs. Then, every vessel recover its FADs. Hence, the utility of each vessel is $2pq$ and the utility of the firm is $(p_f - p)4q - c40$. Even from a theoretical point of view both equilibria are different, in a practical way, both are the same.

5
Conclusiones

Esta Tesis Doctoral ha sido presentada a través de tres ensayos que estudian problemas de optimización logística en entornos dinámicos.

Cada uno de los trabajos presentados muestra algoritmos y técnicas que permiten una mejora de la eficiencia en entornos de alta variabilidad. Aunque las soluciones presentadas son de carácter general, el estudio se ha concretado en el ámbito de la pesca del atún tropical con FADs, pues presenta un entorno de gran complejidad en el que estas técnicas pueden ser aplicadas. Todas las soluciones presentadas son validadas en cada uno de los ensayos a través de simulaciones con datos reales.

En este contexto los dos primeros trabajos presentan soluciones novedosas en el estado del arte. Se combinan sistemas de predicción con técnicas heurísticas para lograr soluciones que permiten una mejora clara en la eficiencia. Los algoritmos propuestos, además de resolver el problema presentado, permiten alcanzar una solución más amplia, válida para entornos dinámicos y estáticos.

Finalmente, el último trabajo realiza un estudio teórico del comportamiento de los barcos atuneros que trabajan con FADs. Este estudio, usando el marco teórico de la Teoría de Juegos, demuestra con datos reales que a través de compensaciones existen escenarios donde todos los agentes ganan. Estas políticas pueden ser aplicables por las empresas para mejorar su eficiencia.

Los tres trabajos presentados nos conducen a través de soluciones para la optimización logística, comenzando por el estudio del caso más particular, en el primero de los ensayos, hasta llegar al caso más general, en el tercero. Más allá de los resultados generales para el avance del conocimiento científico, sin embargo, los resultados de esta tesis tienen también una utilidad práctica para el sector del atún tropical, ámbito del que se extraen los datos para desarrollar las soluciones teóricas propuestas y donde se producen mejoras significativas en términos de eficiencia. Así, debido al carácter generalista de los algoritmos y soluciones desarrolladas, es razonable pensar que los resultados serán igualmente válidos y aplicables en cualquier otro sector donde existan problemas de optimización logística en entornos dinámicos. Además del ahorro directo en distancia y tiempo obtenidos, estos trabajos tienen una aplicación directa hacia la sostenibilidad, pues permiten reducir las emisiones a la atmósfera de una manera importante.

Las líneas futuras son múltiples, entre las más importantes de los dos primeros ensayos están el uso de restricciones (uso de capacidades, ventanas temporales, etc.) o mejorar las predicciones usadas. Respecto al tercer trabajo, sería muy interesante usar el algoritmo desarrollado en el segundo ensayo para mejorar la eficiencia todavía más, así como aplicar la modelización de los agentes en otros escenarios, entre otras.

Bibliografía

Agatz N, Bouman P, Schmidt M (2016) Optimization approaches for the traveling salesman problem with drone. Transportation Science DOI http://dx.doi.org/10.2139/ssrn.2639672 35

Alonso F, Oliveira D, de Souza AZ (2015) Artificial immune systems optimization approach for multiobjective distribution system reconfiguration. IEEE Transactions on Power Systems 30(2):840–847 41

Applegate D, Bixby R, Chvatal V, Cook WJ (2007) The traveling salesman problem: a computational study. Princeton University Press 12

Ariz J, Delgado A, Fonteneau A, Gonzales Costas F, Pallarés P (1992) Logs and tunas in the eastern tropical atlantic. a review of present knowledge and uncertainties. In: Proceedings of the international workshop on fishing for tunas associated with floating objects, La Jolla, CA, pp 21–65 57

Arrizabalaga H, Ariz J, Mina X, de Molina AD, Artetxe I, Pallares P, Iriondo A (2001) Analysis of the activities of supply vessels in the indian ocean from observers data. Doc IOTC 60, 61

Aumann R (1976) Agreeing to disagree. Annals of Statistics 4:1236–1239 61, 69

Avner P, Bruls T, Poras I, Eley L, Gas S, Ruiz P, Wiles M, Sousa-Nunes R, Kettleborough R, Rana A, et al (2001) A radiation hybrid transcript map of the mouse genome. Nature genetics 29(2):194–200 12

Bach P, Dagorn L, Josse E, Bard F, Abbes R, Bertrand A, Misselis C (1998) Experimental research and fish aggregating devices (fads) in french polynesia. SPC Fish Aggregating Device Information Bulletin 3:3–18 13

Bae J, Chung W (2017) A heuristic for a heterogeneous automated guided vehicle routing problem. International Journal of Precision Engineering and Manufacturing 18(6):795–801 33

Baker BM, Ayechew M (2003) A genetic algorithm for the vehicle routing problem. Computers & Operations Research 30(5):787–800 12, 42

Banerjee S (2017) An artificial immune system approach to automated program verification: Towards a theory of undecidability in biological computing. PeerJ Preprints 5:e2690v1 41

Bektas T (2006a) The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega 34(3):209 – 219, DOI http://dx.doi.org/10.1016/j.omega.2004.10.004, URL http://www.sciencedirect.com/science/article/pii/S0305048304001550 35

Bektas T (2006b) The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega 34(3):209–219 35

Belhabib D, Greer K, Pauly D (2018) Trends in industrial and artisanal catch per effort in west african fisheries. Conservation Letters 11(1):e12,360 59

Bernardino HS, Barbosa HJ, Lemonge AC (2007) A hybrid genetic algorithm for constrained optimization problems in mechanical engineering. In: Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, IEEE, pp 646–653 42

Besse PC, Cardot H, Stephenson DB (2000) Autoregressive forecasting of some functional climatic variations. Scandinavian Journal of Statistics 27(4):673–687 15

Bez N, Walker E, Gaertner D, Rivoirard J, Gaspar P (2011) Fishing activity of tuna purse seiners estimated from vessel monitoring system (vms) data. Canadian Journal of Fisheries and Aquatic Sciences 68(11):1998–2010 59

Bjarnadóttir ÁS (2004) Solving the vehicle routing problem with genetic algorithms. PhD Thesis, Technical University of Denmark, Denmark 11, 34, 42

Bland RG, Shallcross DF (1989) Large travelling salesman problems arising from experiments in x-ray crystallography: a preliminary report on computation. Operations Research Letters 8(3):125–128 12

Blum C, Roli A (2003a) Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Comput Surv 35(3):268–308 12, 35

Blum C, Roli A (2003b) Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM computing surveys (CSUR) 35(3):268–308 41

Bonabeau E, Dorigo M, Theraulaz G (1999) Swarm intelligence: From natural to artificial systems. Oxford University Press, Oxford, UK 12

Braekers K, Ramaekers K, Van Nieuwenhuyse I (2016) The vehicle routing problem: State of the art classification and review. Computers & Industrial Engineering 99:300–313 36

Bromhead D, Foster J, Attard R, Findlay J, Kalish J (2003) A Review of the Impact of Fish Aggregating Devices (FADs) on Tuna Fisheries: Final Report to the Fisheries Resources Research Fund. Bureau of Rural Sciences Canberra 57

Carter AE, Ragsdale CT (2006) A new approach to solving the multiple traveling salesperson problem using genetic algorithms. European Journal of Operational Research 175(1):246 – 257, DOI http://dx.doi.org/10.1016/j.ejor.2005.04.027, URL `http://www.sciencedirect.com/science/article/pii/S0377221705004236` 43

Casdagli M (1989) Nonlinear prediction of chaotic time series. Physica D: Nonlinear Phenomena 35(3):335–356 15, 18

Castro JJ, Santiago JA, Santana-Ortega AT (2001) A general theory on fish aggregation to floating objects: an alternative to the meeting point hypothesis. Reviews in fish biology and fisheries 11(3):255–277 14, 37, 57

Cattaruzza D, Absi N, Feillet D, González-Feliu J (2017) Vehicle routing problems for city logistics. EURO Journal on Transportation and Logistics 6(1):51–79 36

Chiang CL (2005) Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. IEEE transactions on power systems 20(4):1690–1699 42

Dantzig G, Fulkerson R, Johnson S (1954) Solution of a large-scale traveling-salesman problem. Journal of the operations research society of America pp 393–410 12

Deng Y, Liu Y, Zhou D (2015) An improved genetic algorithm with initial population strategy for symmetric tsp. Mathematical Problems in Engineering 2015 41

Donald D (2010) Traveling Salesman Problem, Theory and Applications. Intech 11, 12

Dorling K, Heinrichs J, Messier GG, Magierowski S (2017) Vehicle routing problems for drone delivery. IEEE Transactions on Systems, Man, and Cybernetics: Systems 47(1):70–85 35

Du W, Li B (2008) Multi-strategy ensemble particle swarm optimization for dynamic optimization. Information sciences 178(15):3096–3109 41

Duchenne É, Laporte G, Semet F (2007) The undirected m-peripatetic salesman problem: Polyhedral results and new algorithms. Operations research 55(5):949–965 11

Dumas Y, Desrosiers J, Gelinas E, Solomon MM (1995) An optimal algorithm for the traveling salesman problem with time windows. Operations research 43(2):367–371 12

Eksioglu B, Vural AV, Reisman A (2009) The vehicle routing problem: A taxonomic review. Computers & Industrial Engineering 57(4):1472–1483 36

Eyckelhof CJ, Snoek M (2002) Ant systems for a dynamic tsp: Ants caught in a traffic jam. In Ant Algorithms pp 88–99 13

Fiechter CN (1994) A parallel tabu search algorithm for large traveling salesman problems. Discrete Appl Math 51(3):243–267 13

Fonteneau A, Chassot E, Bodin N (2013) Global spatio-temporal patterns in tropical tuna purse seine fisheries on drifting fish aggregating devices (dfads): Taking a historical perspective to inform current challenges⋆. Aquatic Living Resources 26(1):37–48 57

Fonteneau A, Chassot E, Gaertner D (2015) Managing tropical tuna purse seine fisheries through limiting the number of drifting fish aggregating devices in the atlantic: food for thought. Collect Vol Sci Pap ICCAT 71(1):460–475 59

Fonteneau P, Pianet (2000) A worldwide review of purse seine fisheries on fads 40, 57, 59

Food and Agriculture Organization of the United nations, FAO (2012) The state of world fisheries and aquaculture 2012. FAO, Rome 2012 36

Gaertner D, Dreyfus-Leon M (2004) Analysis of non-linear relationships between catch per unit effort and abundance in a tuna purse-seine fishery simulated with artificial neural networks. ICES Journal of Marine Science 61(5):812–820 59

Garcia-Najera A, Bullinaria JA (2011) An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows. Computers & Operations Research 38(1):287–300 12, 35, 42

Garey MR, Johnson DS (1983) Crossing number is np-complete. SIAM Journal on Algebraic Discrete Methods 4(3):312–316 34

Garrido P, Riff MC (2010) Dvrp: a hard dynamic combinatorial optimisation problem tackled by an evolutionary hyper-heuristic. Journal of Heuristics 16(6):795, 834 13

Geng L, Zhang Y, Wang J, Fuh JY, Teo S (2014) Cooperative mission planning with multiple uavs in realistic environments. Unmanned Systems 2(01):73–86 36

Gilman EL (2011) Bycatch governance and best practice mitigation technology in global tuna fisheries. Marine Policy 35(5):590–609 57

Girard C, Benhamou S, Dagorn L (2004) Fad: Fish aggregating device or fish attracting device? a new analysis of yellowfin tuna movements around floating objects. Animal Behaviour 67(2):319–326 58

Golden BL, Levy L, Vohra R (1987) The orienteering problem. Naval Research Logistics (NRL) 34(3):307–318 11

Groba C, Sartal A, Vázquez XH (2015) Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices. Computers & Operations Research 56:22–32 34, 35, 57, 59, 60, 61

Groba C, Sartal A, Vázquez XH (2018) Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels. Engineering Applications of Artificial Intelligence 76:55–66 67

Grötschel M, Padberg MW (1985) Polyhedral theory. The traveling salesman problem pp 251–305 11

Grötschel M, Jünger M, Reinelt G (1991) Optimal control of plotting and drilling machines: a case study. Mathematical Methods of Operations Research 35(1):61–84 12

Guijarro B, Ordines F, Massutí E (2017) Improving the ecological efficiency of the bottom trawl fishery in the western mediterranean: It's about time! Marine Policy 83:204–214 59

Guntsch M, Middendorf M, Schmeck H (2001) An ant colony optimization approach to dynamic tsp 13

Gutin G, Punnen AP (2002) The traveling salesman problem and its variations, vol 12. Springer 11, 12

Hajjam A, Créput JC, Koukam A (2013) From the tsp to the dynamic vrp: An application of neural networks in population based metaheuristic. In Metaheuristics for Dynamic Optimization pp 309–339 13, 34, 35

Hallier JP, Gaertner D (2008) Drifting fish aggregation devices could act as an ecological trap for tropical tuna species. Marine Ecology Progress Series 353:255–264 37

Harsanyi J (1967) Games with incomplete information played by 'bayesian' players. Management Science 14 (3):159–182 61, 69

Helbing D, Tilch B (1998) Generalized force model of traffic dynamics. Phys Rev E 58:133–138, DOI 10.1103/PhysRevE.58.133, URL https://link.aps.org/doi/10.1103/PhysRevE.58.133 35

Helvig CS, Robins G, Zelikovsky A (2003) The moving-target traveling salesman problem. Journal of Algorithms 49(1):153–174 11, 35

Holland J (1975) Adaptation in natural and artificial systems, university of michigan press. Ann Arbor, MI 1(97):5 12, 41

Holland J (1992) Genetic algorithms. Scientific american 267(1):66–73 41

Hospido A, Tyedmers P (2005) Life cycle environmental impacts of spanish tuna fisheries. Fisheries Research 76(2):174–186 59

Howard-Grenville J, Buckle SJ, Hoskins BJ, George G (2014) Climate change and management. Academy of Management Journal 57(3):615–623 53, 68

Huang ZC, Hu XL, Chen SD (2001) Dynamic traveling salesman problem based on evolutionary computation. In: Evolutionary Computation, 2001. Proceedings of the 2001 Congress on, IEEE, vol 2, pp 1283–1288 13

Jeon Y, Reid C, Squires D (2008) Is there a global market for tuna? policy implications for tropical tuna fisheries. Ocean Development & International Law 39(1):32–50 60

Jeong CS, Kim MH (1991) Paper: Fast parallel simulated annealing for traveling salesman problem on simd machines with linear interconnections. Parallel Comput 17(2-3):221–228 13

Jiang Q, Sarker R, Abbass H (2005) Tracking moving targets and the non-stationary traveling salesman problem. Complexity International 11(2005):171–179 33, 35

Jih WR, Hsu Y (2004) A family competition genetic algorithm for the pickup and delivery problems with time window. Bulletin of the College of Engineering, National Taiwan University 90:121–130 11, 34, 42

Jindal P, Kumar A (2011) Multiple target intercepting traveling salesman problem 1 35, 36

Karlaftis MG, Kepaptsoglou K, Sambracos E (2009) Containership routing with time deadlines and simultaneous deliveries and pick-ups. Transportation Research Part E: Logistics and Transportation Review 45(1):210–221 12, 42

Karp RM (1972) Reducibility among combinatorial problems. In: Complexity of computer computations, Springer, pp 85–103 35

Konak A, Coit DW, Smith AE (2006) Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering & System Safety 91(9):992–1007 12

Kosakovsky Pond SL, Posada D, Gravenor MB, Woelk CH, Frost SD (2006) Gard: a genetic algorithm for recombination detection. Bioinformatics 22(24):3096–3098 42

Kuo R, Zulvia FE (2017) Hybrid genetic ant colony optimization algorithm for capacitated vehicle routing problem with fuzzy demand?a case study on garbage collection system. In: Industrial Engineering and Applications (ICIEA), 2017 4th International Conference on, IEEE, pp 244–248 41

Lawler EL, Lenstra JK, Kan R, Shmoys DB (1985) The traveling salesman problem: a guided tour of combinatorial optimization, vol 3. Wiley New York 11

Li C, Yang M, Kang L (2006) A new approach to solving dynamic traveling salesman problems. Simulated Evolution and Learning pp 236–243 13

Li T, Shao G, Zuo W, Huang S (2017) Genetic algorithm for building optimization: State-of-the-art survey. In: Proceedings of the 9th International Conference on Machine Learning and Computing, ACM, pp 205–210 41

Liu CH (2013) The moving-target traveling salesman problem with resupply. Tech. rep., Technical report, The National Chung Cheng University Library. http://ccur. lib. ccu. edu. tw/handle/987654321/7877 33

Liu L, Wang D, Yang S (2009) An immune system based genetic algorithm using permutation-based dualism for dynamic traveling salesman problems. In Applications of Evolutionary Computing pp 725–734 13, 35

Lopez J, Moreno G, Sancristobal I, Murua J (2014) Evolution and current state of the technology of echo-sounder buoys used by spanish tropical tuna purse seiners in the atlantic, indian and pacific oceans. Fisheries Research 155:127–137 57, 58, 59

Lynn N, Suganthan PN (2017) Ensemble particle swarm optimizer. Applied Soft Computing 55:533–548 41

Madau FA, Furesi R, Pulina P (2018) The technical efficiency in sardinian fisheries cooperatives. Marine Policy 95:111–116 60

Marcellino M, Stock JH, Watson MW (2006) A comparison of direct and iterated multistep ar methods for forecasting macroeconomic time series. Journal of Econometrics 135(1):499–526 16

Maufroy A, Kaplan DM, Bez N, De Molina AD, Murua H, Floch L, Chassot E, editor: Jan Jaap Poos H (2016) Massive increase in the use of drifting fish aggregating devices (dfads) by tropical tuna purse seine fisheries in the atlantic and indian oceans. ICES Journal of Marine Science 74(1):215–225 57, 59

Mavrovouniotis M, Yang S (2013) Ant colony optimization with immigrants schemes for the dynamic travelling salesman problem with traffic factors. Applied Soft Computing 13(10):4023–4037 41

Ménard F, Stéquert B, Rubin A, Herrera M, Marchal É (2000) Food consumption of tuna in the equatorial atlantic ocean: Fad-associated versus unassociated schools. Aquatic living resources 13(4):233–240 57, 58

Menezes MB, Ketzenberg M, Oliva R, Metters R (2015) Service delivery to moving demand points using mobile servers. International Journal of Production Economics 168:158–166 33, 34

Menezes T, Tedesco P, Ramalho G (2006) Negotiator agents for the patrolling task. Advances in Artificial Intelligence-IBERAMIA-SBIA 2006 pp 48–57 36

Miyake M, Guillotreau P, Sun C, Ishimura G (2010) Recent developments in the tuna industry. FAO, Rome 2012 36, 37

Montoya-Torres JR, Franco JL, Isaza SN, Jiménez HF, Herazo-Padilla N (2015) A literature review on the vehicle routing problem with multiple depots. Computers & Industrial Engineering 79:115–129 36

Moon C, Kim J, Hur S (2002) Integrated process planning and scheduling with minimizing total tardiness in multi-plants supply chain. Computers & Industrial Engineering 43(12):331–349 13

Moreno G, Dagorn L, Sancho G, Itano D (2007) Fish behaviour from fishers'knowledge: the case study of tropical tuna around drifting fish aggregating devices (dfads). Canadian Journal of Fisheries and Aquatic Sciences 64(11):1517–1528 14, 37, 58

Morón J, Areso JJ, Pallares P (2001) Statistics and technical information about the spanish purse seine fleet in the pacific. In: 14th Meeting of the Standing Committee on Tuna and Billfish, Noumea, New Caledonia, pp 9–16 58

Nash J (1951) Non-cooperative games. The Annals of Mathematics 54 (2):286–295 61, 69

Osman IH, Laporte G (1996) Metaheuristics: A bibliography 41

Özgökmen TM, Griffa A, Mariano AJ, Piterbarg LI (2000) On the predictability of lagrangian trajectories in the ocean. Journal of Atmospheric and Oceanic Technology 17(3):366–383 14, 39

Pantrigo JJ, Duarte A (2013) Low-level hybdridization of scatter search and particle filter for dynamic tsp solving. Metaheuristic for Dynamic Optimization pp 291–308 11, 13

Parker RW, Vázquez-Rowe I, Tyedmers PH (2015) Fuel performance and carbon footprint of the global purse seine tuna fleet. Journal of Cleaner Production 103:517–524 37, 53, 59

Pérez J (2004) A hybrid approach for a constraint routing problem. In: Proceedings of the Fourth International Conference on Hybrid Intelligent Systems, Kitakyushu, Japan 12

Potvin JY (1996) Genetic algorithms for the traveling salesman problem. Annals of Operations Research 63(3):337–370 12, 42

Psaraftis HN (1988) Dynamic vehicle routing problems. Vehicle Routing: Methods and Studies (16):223–248 13

Pushkarini Agharkar SDB, Bullo F (2015) Vehicle routing algorithms for radially escaping targets. SIAM Journal on Control and Optimization 53:21 35

Rajeswari G (2009) Fish aggregating devices 36

Razali NM, Geraghty J, et al (2011) Genetic algorithm performance with different selection strategies in solving tsp. In: Proceedings of the world congress on engineering, International Association of Engineers Hong Kong, vol 2, pp 1134–1139 42

Reinelt G (1994) The traveling salesman: computational solutions for TSP applications. Springer-Verlag 12, 15, 42

Ruan Q, Zhang Z, Miao L, Shen H (2013) A hybrid approach for the vehicle routing problem with three-dimensional loading constraints. Computers & Operations Research 40(6):1579–1589 12, 42

Rust S, Yamazaki S, Jennings S, Emery T, Gardner C (2017) Excess capacity and efficiency in the quota managed tasmanian rock lobster fishery. Marine Policy 76:55–62 59

Salas S, Gaertner D (2004) The behavioural dynamics of fishers: management implications. Fish and fisheries 5(2):153–167 59

Shetty VK, Sudit M, Nagi R (2008) Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles. Computers & Operations Research 35(6):1813–1828 36

Shima T, Schumacher C (2005) Assignment of cooperating uavs to simultaneous tasks using genetic algorithms. In: Proc. AIAA Guidance, Navigation, and Control Conference and Exhibit. San Francisco 36

Shuai Y, Bradley S, Shoudong H, Dikai L (2013) A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. European Journal of Operational Research 228(1):72 – 82, DOI http://dx.doi.org/10.1016/j.ejor.2013.01.043, URL http://www.sciencedirect.com/science/article/pii/S0377221713000908 45

Simões A, Costa E (2011) Chc-based algorithms for the dynamic traveling salesman problem. Applications of Evolutionary Computation pp 354–363 13

Smith GC, Smith S (2002) An enhanced genetic algorithm for automated assembly planning. Robotics and Computer-Integrated Manufacturing 18(5):355–364 12, 42

Song CH, Lee K, Lee WD (2003) Extended simulated annealing for augmented tsp and multi-salesmen tsp. In: Neural Networks, 2003. Proceedings of the International Joint Conference on, IEEE, vol 3, pp 2340–2343 41

Srinivas M, Patnaik LM (1994) Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man, and Cybernetics 24(4):656–667 42, 44

Stieber A, Fügenschuh A (2017) The multiple traveling salesmen problem with moving targets and nonlinear trajectories 33, 34, 35

Stieber A, Fügenschuh A, Yuan Z (2015) School Taxi Routing for Children with Special Needs. Helmut-Schmidt-Univ., Univ. der Bundeswehr Hamburg 33, 35, 36

Sundar K, Venkatachalam S, Rathinam S (2017) Analysis of mixed-integer linear programming formulations for a fuel-constrained multiple vehicle routing problem. Unmanned Systems pp 1–11 33

Thomas N, Poongodi DP (2009) Position control of dc motor using genetic algorithm based pid controller. In: Proceedings of the World Congress on Engineering, London, UK, vol 2, pp 1–3 42

Toth P, Vigo D (2014) Vehicle routing: problems, methods, and applications. SIAM 36

Tyedmers P, Parker R (2012) Fuel consumption and greenhouse gas emissions from global tuna fisheries: A preliminary assessment. Tech. rep., ISSF Technical Report 2012-03. International Seafood Sustainability Foundation, McLean, Virginia, USA 12

Venkatesh P, Singh A (2015) Two metaheuristic approaches for the multiple traveling salesperson problem. Applied Soft Computing 26:74 – 89, DOI http://dx.doi.org/10.1016/j.asoc.2014.09.029, URL http://www.sciencedirect.com/science/article/pii/S1568494614004827 35

Wang Hf, Wang Dw, Yang Sx (2007) Evolutionary algorithms in dynamic environments. Control and Decision 22(2):127 41

Wang X, Regan AC (2002) Local truckload pickup and delivery with hard time window constraints. Transportation Research Part B: Methodological 36(2):97–112 12, 42

Winter G, Periaux J, Galan M, Cuesta P (1996) Genetic algorithms in engineering and computer science. John Wiley & Sons, Inc. 12, 42

Yan XS, Liu HM, Yan J, Wu QH (2007) A fast evolutionary algorithm for traveling salesman problem. In Proceedings of the 3rd International Conference on Natural Computation, 2007 ICNC 2007 4:85–90 13

Yang R (1997) Solving large travelling salesman problems with small populations. Second International Conference on Genetic Algorithms in Engineering Systems pp 157–162 21, 22, 43

Yegnanarayana B (2009) Artificial neural networks. PHI Learning Pvt. Ltd. 41

Younes A, Basir O, Calamai P (2003) A benchmark generator for dynamic optimization. In Proceedings of the 3rd WSEAS International Conference on Soft Computing, Optimization, Simulation & Manufacturing Systems, Malta 13

Zamir S (2013) Bayesian Games: Games with Incomplete Information. In: Meyers R. (eds) Encyclopedia of Complexity and Systems Science. Springer 69

Zeller D, Cashion T, Palomares M, Pauly D (2018) Global marine fisheries discards: a synthesis of reconstructed data. Fish and Fisheries 19(1):30–39 57

Zhang GP, Qi M (2005) Neural network forecasting for seasonal and trend time series. European journal of operational research 160(2):501–514 16

Zhou A, Kang L, Yan Z (2003) Solving dynamic tsp with evolutionary approach in real time. In: Evolutionary Computation, 2003. CEC'03. The 2003 Congress on, IEEE, vol 2, pp 951–957 13, 34, 41